

Perpustakaan SKTM

Nama: Nor Azila Mohd Bukhari
No. matrik: WEK 000298
Tajuk: Web Calculator
Penyelia: Puan Azwina Mohd Yusof
Moderator: Prof. Madya Dr Siti Salwah Salim

Abstrak

Laporan ini adalah untuk menentukan pembangunan kalkulator berasaskan web yang dinamakan *Web Calculator*. Terdapat lapan bab di dalam laporan ini.

Bab yang pertama adalah pengenalan kepada maklumat projek iaitu berkaitan dengan latarbelakang, kekurangan sistem sedia ada, objektif yang ingin dicapai, skop projek, hasil yang dijangkakan dan perancangan projek.

Bab yang kedua akan memaparkan keputusan kajian literasi. Ulasan tentang kalkulator yang sedia ada dan peralatan pembangunan dimasukkan di sini. Bab yang ketiga pula memaparkan pemilihan metodologi di dalam pembangunan perisian dan teknik pengumpulan maklumat yang dilakukan.

Bab yang keempat memperlihatkan keperluan di dalam pembangunan perisian ini dan juga bahasa pengaturcaraan yang dipilih. Bab yang kelima menunjukkan rajah *use case* dan carta alir bagi sistem ini. Rekabentuk antaramuka juga dimasukkan di bahagian ini.

Bab seterusnya adalah pembangunan sistem. Di dalam bab ini, cara sistem dibangunkan diterang dengan jelas termasuk peralatan perisian dan perkakasan yang digunakan. Kemudian, bab ketujuh menerangkan bagaimana pengujian sistem dilakukan dan teknik-teknik pengujian yang dipilih. Bab yang terakhir menunjukkan perbincangan bagi sistem yang telah dibangunkan.

Penghargaan

Dengan nama Allah Yang Maha Pemurah lagi Maha Mengasihani. Bersyukur kehadiran Ilahi, atas segala limpahan kurnia-Nya, dan kerana dengan izin-Nya, saya dapat menyiapkan laporan dan projek latihan ilmiah saya.

Ribuan terima kasih buat Pn. Azwina Mohd Yusof selaku penyelia, di atas segala nasihat, bantuan, pandangan galakan dan semangat, serta tunjuk ajar sepanjang saya menyiapkan projek latihan ilmiah ini. Tidak lupa juga kepada moderator saya Prof. Madya Dr Siti Salwah Salim, kerana sudi meluangkan masa beliau dan atas pandangan serta bantuan yang telah diberikan.

Di kesempatan ini juga, saya ingin merakamkan penghargaan dan ucapan terima kasih kepada kedua ibu bapa saya, yang sentiasa memberi dorongan dan sokongan yang tidak terhingga. Tidak lupa buat rakan-rakan seperjuangan dan akhir sekali buat semua yang terlibat sama ada secara langsung atau tidak langsung sepanjang menjalani dan menyiapkan latihan ilmiah ini.

Sekian, terima kasih.

Senarai Isi Kandungan

Pengenalan

1.1	Pengenalan	1
1.2	Kekurangan	1
1.3	Objektif	2
1.4	Skop	3
1.5	Hasil yang Dijangkakan	3
1.6	Perancangan Projek	3
1.7	Organisasi Bab	5

Kajian Literasi

2.1	Kalkulator	6
2.2	Fungsi Kalkulator	6
2.3	Sistem Nombor	9
2.3.1	Sistem Nombor Binari	9
2.3.1.1	Penukaran nombor desimal ke nombor binari	11
2.3.1.2	Penukaran nombor binari ke nombor desimal	11
2.3.1.3	Penambahan nombor binari	12
2.3.1.4	Penolakan nombor binari	12
2.3.1.5	Pendaraban nombor binari	13
2.3.1.6	Pembahagian nombor binari	14
2.3.2	Sistem Nombor Oktal	15
2.3.2.1	Penukaran nombor desimal ke nombor oktal	15
2.3.2.2	Penukaran nombor oktal ke nombor desimal	16
2.3.2.3	Penambahan nombor oktal	16
2.3.2.4	Penolakan nombor oktal	17
2.3.3	Sistem Nombor Heksadesimal	20
2.3.3.1	Penukaran nombor desimal ke nombor heksadesimal	21
2.3.3.2	Penukaran nombor heksadesimal ke nombor desimal	22
2.3.3.3	Penambahan nombor heksadesimal	22
2.3.3.4	Penolakan nombor heksadesimal	23
2.4	Web Calculator	24
2.5	Kalkulator Binari, Oktal, Desimal dan Heksadesimal	24
2.5.1	Stand-alone	25
2.5.2	Berasaskan web (web based)	27
2.6	Aplikasi Web	29
2.7	Kajian Bahasa Pengaturcaraan	30
2.7.1	Java	30
2.7.2	C++	34
2.8	Kajian Metodologi	36
2.8.1	Model Air Terjun	36
2.8.2	Model Air Terjun dengan Prototaip	36
2.8.3	Model V	37

2.8.4	Model Prototaip	37
2.8.5	Fasa Pembangunan: Penokokan dan Iterasian	38
2.9	Rumusan	39
Metodologi		
3.1	Pemilihan Metodologi	40
3.2	Teknik Pengumpulan Maklumat	41
3.3	Rumusan	42
Analisa Sistem		
4.1	Keperluan Fungsian	43
4.2	Keperluan Bukan Fungsian	43
4.3	Pemilihan Bahasa Pengaturcaraan	44
4.4	Keperluan Perkakasan	46
4.5	Keperluan Perisian	47
4.6	Rumusan	48
Rekabentuk Sistem		
5.1	Pendahuluan	49
5.2	Model Use Case	49
5.3	Carta Alir	51
5.4	Rekabentuk Antaramuka	52
Pembangunan Sistem		
6.1	Pengenalan	58
6.2	Dasar Pelaksanaan	58
6.3	Persekitaran Pembangunan	58
6.3.1	Spesifikasi Perkakasan	59
6.3.2	Peralatan Perisian	60
6.4	Pengkodan	61
6.4.1	Pendekatan Pengkodan	61
6.4.2	Gaya Pengkodan	65
6.4.3	Pengkodan <i>Web Calculator</i>	65
6.5	Metodologi yang Digunakan	66
Pengujian Sistem		
7.1	Pengenalan	67
7.2	Objektif Pengujian	68
7.3	Konsep Pengujian	69
7.4	Asas Pengujian	70
7.5	Strategi Pengujian	72

7.5.1	Pengujian Unit	72
7.6	Pengujian Integrasi	74
7.7	Pengujian Sistem	75
7.8	Teknik Pengujian	76
7.8.1	Pengujian Kotak Putih	76
7.8.2	Pengujian Kotak Hitam	77
Perbincangan		
8.1	Pendahuluan	78
8.2	Kekuatan Sistem	78
8.3	Kekangan Sistem	80
8.4	Masalah dan Penyelesaian	81
8.5	Peningkatan Masa Hadapan	83
8.6	Pengetahuan yang Didapati	84
8.7	Kesimpulan	84
Apendiks A: Borang Pengujian Unit		85
Apendiks B: Borang Penerimaan Pengguna		86
Apendiks C: Manual Pengguna		87
Rujukan		95
Bibliografi		96

Senarai Jadual

Jadual 2.1: 20 nombor binari pertama dan nilai desimal yang sepadan	10
Jadual 2.2: 18 nombor oktal dan nombor desimal yang sepadan	15
Jadual 2.3: 16 nombor binari, heksadesimal dan desimal yang sepadan	20
Jadual 6.1: Spesifikasi perkakasan	59
Jadual 6.2: Peralatan perisian untuk pembangunan	60
Jadual 2.4: Kalendar binari untuk 2020 hingga 2024	21
Jadual 2.5: Senarai nombor binari	24
Jadual 2.6: Senarai nombor oktal	25
Jadual 2.7: Senarai nombor binari, heksadesimal dan desimal yang sepadan	26
Jadual 2.8: Model nombor binari yang setiap model mempunyai nombor binari	27
Jadual 2.9: Model nombor binari yang setiap model mempunyai nombor binari	28
Jadual 2.10: Model nombor binari yang setiap model mempunyai nombor binari	29
Jadual 2.11: Model nombor binari yang setiap model mempunyai nombor binari	30
Jadual 2.12: Model nombor binari yang setiap model mempunyai nombor binari	31
Jadual 2.13: Model nombor binari yang setiap model mempunyai nombor binari	32
Jadual 2.14: Model nombor binari yang setiap model mempunyai nombor binari	33
Jadual 2.15: Model nombor binari yang setiap model mempunyai nombor binari	34
Jadual 2.16: Model nombor binari yang setiap model mempunyai nombor binari	35
Jadual 2.17: Model nombor binari yang setiap model mempunyai nombor binari	36
Jadual 2.18: Model nombor binari yang setiap model mempunyai nombor binari	37
Jadual 2.19: Model nombor binari yang setiap model mempunyai nombor binari	38
Jadual 2.20: Model nombor binari yang setiap model mempunyai nombor binari	39

Senarai Rajah

Rajah 1.1: Carta Gantt	4
Rajah 2.1: Contoh-contoh kalkulator yang terdapat di pasaran	8
Rajah 2.2: Kalkulator di dalam sistem pengoperasian Windows	25
Rajah 2.3: Kalkulator di dalam sistem pengoperasian Linux	26
Rajah 2.4: Kalkulator berasaskan web MrCalculator's Hexadecimal Calculator	27
Rajah 2.5: Kalkulator berasaskan web Javascript Calculator	28
Rajah 2.6: Senibina aplikasi web	30
Rajah 3.1: Model air terjun dengan prototaip	40
Rajah 5.1: Model <i>use case</i> bagi keseluruhan sistem	49
Rajah 5.2: Model <i>use case</i> bagi setiap modul sistem nombor	50
Rajah 5.3: Carta alir bagi keseluruhan sistem	51
Rajah 5.4: Rekabentuk antaramuka utama	52
Rajah 5.5: Rekabentuk antaramuka kalkulator sistem nombor binari	53
Rajah 5.6: Rekabentuk antaramuka kalkulator sistem nombor oktal	54
Rajah 5.7: Rekabentuk antaramuka kalkulator sistem nombor desimal	55
Rajah 5.8: Rekabentuk antaramuka kalkulator sistem nombor heksadesimal	56
Rajah 5.9: Rekabentuk antaramuka tettingkap bantuan	57
Rajah 7.1: Peringkat-peringkat pengujian	72

Pengenalan

1.1 Pengenalan

Projek ini adalah mengenai kalkulator berasaskan web. Kalkulator ini boleh melaksanakan pengiraan dalam sistem nombor binari, oktal, desimal dan heksadesimal. Operasi-operasi asas boleh dilakukan ke atas sistem-sistem nombor ini iaitu penambahan, penolakan, pendaraban dan pembahagian. Terdapat operasi tambahan bagi sistem nombor binari iaitu operasi AND, OR, NOT dan XOR. Kalkulator ini dibina berasaskan web kerana untuk memudahkan pengguna internet menggunakannya secara atas talian (*online*). Kalkulator ini sesuai untuk pengguna yang menggunakan sistem-sistem nombor ini dan tidak mempunyai kalkulator seumpama ini di dalam pakej sistem pengoperasian mereka.

1.2 Kekurangan

Projek ini dimotivasikan melalui penemuan beberapa kelemahan di dalam beberapa kalkulator berasaskan web yang wujud. Kelemahan utama yang dikenalpasti adalah tidak menyediakan kemudahan pengiraan untuk keempat-empat sistem nombor iaitu sistem nombor binari, oktal, desimal dan heksadesimal. Kalkulator yang sedia ada hanya menyediakan kemudahan pengiraan terhadap salah satu atau dua sistem nombor tersebut sahaja.

Selain itu, susunan butang-butang kalkulator yang sedia ada juga tidak tersusun. Ini kerana saiz setiap butang berbeza. Susunan nombor-nombor dan operasi-operasi ini juga tidak mengikut piawai. Ini mungkin akan mengelirukan pengguna yang telah biasa dengan susunan yang sering digunakan.

1.3 Objektif

Objektif bagi sistem ini ialah:-

- a) Menyediakan kalkulator kepada pengguna internet

Sistem yang sedia ada adalah *stand-alone* dan berasaskan *web*. Walaupun kalkulator seumpama ini wujud secara *stand-alone* iaitu melalui pakej sistem pengoperasian seperti Windows dan Linux, namun bagi pengguna Macintosh, tiada kalkulator seumpama ini. Maka sistem ini dapat membantu pengguna Macintosh dalam menjalankan pengiraan berdasarkan sistem-sistem nombor ini.

- b) Menghasilkan sistem yang lebih mesra pengguna

Sistem yang sedia ada agak mengelirukan pengguna terutama bagi pengguna yang pertama kali menggunakan sistem tersebut dan pengguna yang kurang faham tentang sistem-sistem nombor tersebut. Sistem yang akan dibangunkan lebih mesra-pengguna yang mana ia dapat mengurangkan kekeliruan pengguna seterusnya memudahkan latihan bagi pengguna yang pertama kali menggunakan sistem seumpama ini.

- c) Menghasilkan sistem yang dapat mengurangkan penggunaan kertas dan pen dalam pengiraan

Melalui penghasilan kalkulator ini, penggunaan kertas dan pen dapat dikurangkan atau dihapus dalam melakukan pengiraan terhadap sistem-sistem nombor ini.

1.4 Skop

Kalkulator ini akan menyokong pengiraan terhadap empat sistem nombor iaitu sistem nombor binari, oktal, desimal dan heksadesimal. Semua sistem nombor dapat menjalankan operasi-operasi penambahan, penolakan, pendaraban, pembahagian dan operasi Boolean AND, OR, NOT dan XOR kecuali sistem nombor desimal sahaja yang dapat menjalankan operasi-operasi asas. Selain itu terdapat juga menu bantuan (*help*) bagi membantu pengguna menggunakan kalkulator ini.

1.5 Hasil yang Dijangkakan





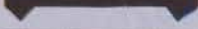

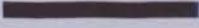


Hasil yang dijangkakan bagi sistem ini adalah seperti berikut:

- sistem boleh dilarikan (*run*): boleh melaksanakan semua operasi-operasi yang disediakan terhadap sistem nombor binari, oktal, desimal dan heksadesimal.
- Ramah pengguna dengan antaramuka grafik pengguna

1.6 Perancangan Projek

Carta Gantt adalah cara yang mudah untuk merancang tugas. Ia adalah carta yang diwakili oleh palang sebagai tugas atau aktiviti. Kapanjangan setiap palang mewakili tempoh tugas dilaksanakan. Rajah di sebelah adalah carta Gantt yang digunakan untuk merancang projek ini.

ID	Task Name	Duration	Start	Finish	Qtr 3, 2002							Qtr 4, 2002			Qtr 1, 20
					Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan			
1	Kajian Literasi	3 wks	Mon 7/1/02	Fri 7/19/02											
2	Analisis Keperluan	3 wks	Mon 7/22/02	Fri 8/9/02											
3	Rekabentuk Sistem	2 wks	Mon 8/12/02	Fri 8/23/02											
4	Pemprototaipan Sistem	4 wks	Mon 8/26/02	Fri 9/20/02											
5	Pembangunan	10 wks	Mon 9/23/02	Fri 11/29/02											
6	Pengujian dan Penyelenggara	4 wks	Fri 11/22/02	Thu 12/19/02											
7	Dokumentasi	30 wks	Mon 7/1/02	Fri 1/24/03											

Project: Project2 Date: Thu 2/20/03	Task		Milestone		External Tasks	
	Split		Summary		External Milestone	
	Progress		Project Summary		Deadline	

Rajah 1.1: Carta Gantt

1.7 Organisasi Bab

Bab 1 – Bab ini memperkenalkan projek yang akan dibangunkan

Bab 2 – Bab ini memaparkan keputusan kajian literasi yang dijalankan terhadap dua bahagian iaitu sistem yang berkaitan dan bahasa pengaturcaraan yang digunakan.

Bab 3 – Bab ini menghuraikan metodologi yang digunakan dalam membangunkan sistem ini dan teknik-teknik yang digunakan dalam mengumpulkan maklumat.

Bab 4 – Bab ini menghuraikan keperluan-keperluan yang diperlukan iaitu keperluan fungsian, keperluan bukan fungsian, keperluan perkakasan dan keperluan perisian.

Bab 5 – Bab ini memaparkan rekabentuk sistem.

KAJIAN LITERASI

2.1 Kalkulator

[1] Kalkulator adalah satu peranti elektronik yang kecil yang mana operasi aritmetik boleh dilakukan ke atas nombor yang dimasukkan melalui satu papan kekunci. Penyelesaian terakhir dan nombor-nombor secara amnya dipaparkan ke atas LCD. Julat kalkulator adalah dari peranti yang paling mudah yang boleh menjalankan operasi asas aritmetik ke peranti yang boleh menjalankan manipulasi matematik dan statistik yang sofistikated dan ia mungkin diprogramkan melalui banyak langkah. Modul ingatan tambahan mengandungi set-set istimewa program untuk bidang-bidang tertentu - sebagai contoh kejuruteraan, navigasi atau perniagaan – mungkin dijual sebagai aksesori kepada kalkulator yang lebih mahal.

2.2 Fungsi Kalkulator

Secara amnya, kalkulator digunakan dalam pengiraan. Pada masa kini, Kalkulator sering digunakan untuk menyelesaikan pelbagai masalah berkaitan dengan matematik. Terdapat enam jenis kalkulator di pasaran iaitu:

1. *Desktop*

Kalkulator ini membolehkan penggunaanya menjalankan pengiraan yang tidak kompleks. Ia sama seperti kalkulator biasa yang lain cuma bentuknya sahaja yang berbeza. Bentuknya membolehkan pengguna meletakkannya di atas meja. Kalkulator jenis ini sesuai untuk aplikasi perniagaan dan penggunaan persendirian.

2. *Printer*

Kalkulator ini sesuai untuk pelbagai jenis perniagaan, membolehkan cetakan dilakukan dengan pantas di dalam dua warna.

3. *Handheld*

Jenis ini pula boleh melakukan pengiraan yang tidak kompleks seperti penambahan, penolakan, pendaraban dan pembahagian. Fungsinya sama dengan kalkulator *desktop* tetapi bentuknya sahaja yang kecil. Ini membolehkan pengguna memegangnya dengan mudah dan dibawa ke mana-mana.

4. *Saintifik*

Kalkulator jenis ini boleh menyelesaikan masalah matematik yang kompleks. Antara pengiraan yang boleh dilakukan ialah sin, kos, tan, log, faktorial dan sebagainya.



Desktop



Handheld



Printer



Saintifik

Rajah 2.1: Contoh-contoh kalkulator yang terdapat di pasaran

2.3 Sistem Nombor

Sistem nombor menakrifkan satu set nilai yang digunakan untuk mewakili kuantiti. Sistem nombor yang digunakan sekarang adalah sistem Arab (Arabic Sistem). Ia pertama kali dibangunkan oleh orang Hindu dan telah digunakan seawal abad ke-3 SM. Nilai dasar sistem nombor adalah bilangan nilai yang berbeza yang dipunyai oleh set tersebut sebelum ia mengulang dirinya sendiri. Sebagai contoh, desimal mempunyai dasar nilai 10, 0 hingga 9.

- Binari = 2 (0, 1)
- Oktal = 8 (0 - 7)
- Desimal = 10 (0 - 9)
- Duodesimal = 12 (digunakan dalam sesetengah hal oleh orang Roman)
- Heksadesimal = 16 (0 - 9, A-F)
- Vigesimal = 20 (digunakan oleh orang Mayan)
- Seksagesimal = 60 (digunakan oleh orang Babylon)

2.3.1 Sistem Nombor Binari

[2] Pada kebiasaannya, sistem nombor binari adalah sistem nombor dengan dasar 2. Sistem nombor ini adalah yang sering digunakan di dalam komputer. Satu digit binari (atau bit) sama ada 0 atau 1. Perwakilan nombor oleh digit binari dipanggil notasi binari.

Sistem binari juga digunakan untuk menggambarkan mana-mana sistem yang mempunyai dua keadaan yang mungkin. Sebagai contoh,

setiap elemen yang mengandungi di dalam mana-mana komputer adalah satu sistem binari. Satu keadaan digunakan untuk menandakan digit binari 0 dan yang satu lagi untuk menandakan digit binari 1. Lazimnya, elemen storan atau unit maklumat dalam mana-mana sistem binari dirujuk sebagai 1 bit. Berikut ialah 20 nombor binari yang pertama.

Jadual 2.1: 20 nombor binari pertama dan nilai desimal yang sepadan

Desimal	Binari	Desimal	Binari
1	= 1	11	= 1011
2	= 10	12	= 1100
3	= 11	13	= 1101
4	= 100	14	= 1110
5	= 101	15	= 1111
6	= 110	16	= 10000
7	= 111	17	= 10001
8	= 1000	18	= 10010
9	= 1001	19	= 10011
10	= 1010	20	= 10100

Untuk menjelaskan nilai nombor binari, maka, $a_{n-1} 2^{n-1} + a_{n-2} 2^{n-2} + \dots + a_0$ adalah diwakili sebagai $a_{n-1} a_{n-2} \dots a_0$, yang mana a_i adalah sama ada 1 atau 0 dan n adalah bilangan digit dari kiri titik binari (radiks). Berikut adalah contoh penukaran nombor desimal ke binari dan sebaliknya.

2.3.1.1 Penukaran nombor desimal ke nombor binari

Penukaran nombor desimal ke nombor binari dilakukan dengan melakukan pembahagian berulang.

Contoh:

$$125 = 11111011_2$$

$$\begin{array}{r|l} 2 & 125 \\ \hline 2 & 62 \dots 1 \\ \hline 2 & 31 \dots 0 \\ \hline 2 & 15 \dots 1 \\ \hline 2 & 7 \dots 1 \\ \hline 2 & 3 \dots 1 \\ \hline & 1 \dots 1 \end{array}$$

2.3.1.2 Penukaran nombor binari ke nombor desimal

Penukaran nombor binari ke nombor desimal pula dilakukan dengan menjumlahkan hasil darab digit nombor binari dengan 2^{n-1} yang mana n adalah bilangan digit nombor binari tersebut.

Contoh: $101_2 = ?$

$$\begin{aligned} 101_2 &= 1 \times 2^{3-1} + 0 \times 2^{3-2} + 1 \times 2^{3-3} \\ &= 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 \\ &= 4 + 1 \\ &= 5 \end{aligned}$$

2.3.1.3 Penambahan nombor binari

Penambahan nombor binari adalah sama dengan penambahan nombor desimal. Sebenarnya aritmetik binari lebih mudah dipelajari. Jadual lengkap untuk penambahan binari adalah seperti berikut:

$$0 + 0 = 0$$

$$0 + 1 = 1$$

$$1 + 0 = 1$$

$$1 + 1 = 0$$

Pembawa (*carry*) adalah sama dalam aritmetik desimal. Disebabkan 1 adalah digit yang terbesar dalam sistem binari, maka mana-mana hasil tambah yang lebih daripada 1 memerlukan digit tersebut dibawa ke atas.

Contoh:

$$\begin{array}{r} 11111 \\ + 10100 \\ \hline 100011 \end{array}$$

2.3.1.4 Penolakan nombor binari

Penolakan adalah pembalikan daripada operasi penambahan. Dalam aritmetik binari, seperti desimal, ia sesuai jika penolakan digit yang lebih besar dari digit yang lebih kecil. Satu-satunya kes yang mana ianya berlaku dengan nombor binari

bila 1 ditolak dari 0. Bakinya adalah satu tetapi ia adalah pinjaman 1 dari kolum sebelah kiri. Berikut adalah jadual penolakan binari.

$$0 - 0 = 0$$

$$1 - 0 = 1$$

$$1 - 1 = 0$$

$$0 - 1 = 1 \text{ dengan pinjaman 1}$$

Contoh:

$$\begin{array}{r} 10000 \\ - \quad 11 \\ \hline 1101 \end{array}$$

2.3.1.5 Pendaraban nombor binari

Berikut adalah jadual untuk pendaraban:

$$0 \times 0 = 0$$

$$1 \times 0 = 0$$

$$0 \times 1 = 0$$

$$1 \times 1 = 1$$

Contoh:

$$\begin{array}{r}
 1100 \\
 \times 1010 \\
 \hline
 0000 \\
 1100 \\
 -0000 \\
 \hline
 1100 \\
 \hline
 1111000
 \end{array}$$

2.3.1.6 Pembahagian nombor binari

Pembahagian nombor binari adalah mudah seperti dalam sistem nombor desimal (atau lain-lain), pembahagian oleh digit 0 adalah tidak memberi apa-apa makna. Jadual lengkap adalah:

$$0 \div 1 = 0$$

$$1 \div 1 = 1$$

Contoh:

$$\begin{array}{r}
 101 \overline{) 11001} \\
 \underline{101} \\
 101 \\
 \underline{101} \\
 101 \\
 \underline{101} \\
 0000
 \end{array}$$

2.3.2 Sistem nombor oktal

[2] Sistem nombor oktal mempunyai dasar atau radiks 8; 8 simbol berbeza digunakan untuk mewakili nombor. Simbol-simbol ini adalah 0, 1, 2, 3, 4, 5, 6, dan 7. Jadual berikut menunjukkan 18 nombor oktal pertama dan nombor desimal yang sepadan.

Jadual 2.2: 18 nombor oktal dan nombor desimal yang sepadan.

Oktal	Desimal	Oktal	Desimal
0	= 0	11	= 9
1	= 1	12	= 10
2	= 2	13	= 11
3	= 3	14	= 12
4	= 4	15	= 13
5	= 5	16	= 14
6	= 6	17	= 15
7	= 7	20	= 16
10	= 8	21	= 17

2.3.2.1 Penukaran nombor desimal ke nombor oktal.

Penukaran nombor desimal ke nombor oktal boleh dilakukan dengan melakukan pembahagian berulang.

$$200 = 310_8$$

$$\begin{array}{r|l} 8 & 200 \\ \hline 8 & 25 \dots 0 \\ & 3 \dots 1 \end{array}$$

2.3.2.2 Penukaran nombor oktal ke nombor desimal

Penukaran nombor oktal ke nombor desimal pula boleh dilakukan dengan menjumlahkan hasil darab digit nombor oktal dengan 8^{n-1} yang mana n adalah posisi digit nombor oktal tersebut.

Contoh:

$$\begin{aligned} 7574_8 &= 7_{10} \times 8^3_{10} + 5_{10} \times 8^2_{10} + 7_{10} \times 8_{10} + 4_{10} \\ &= 7_{10} \times 512_{10} + 5_{10} \times 64_{10} + 7_{10} \times 8_{10} + 4_{10} \\ &= 3584_{10} + 320_{10} + 56_{10} + 4_{10} \\ &= 3964_{10} \end{aligned}$$

2.3.2.3 Penambahan nombor oktal

Penambahan nombor oktal tidak susah, hanya perlu mengingat apabila hasil tambah dua digit melebihi 7, satu pembawa akan terhasil. Bandingkan dua contoh di bawah:

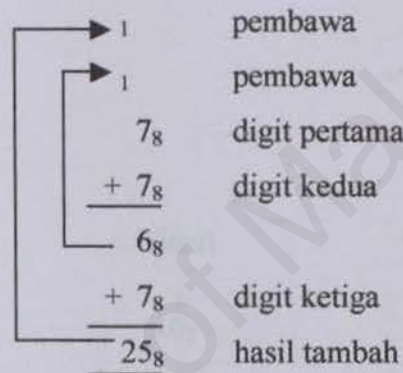
$$\begin{array}{r} 4_8 \\ + 2_8 \\ \hline 6_8 \\ \hline \end{array}$$

$$\begin{array}{r} 4_8 \\ + 4_8 \\ \hline 10_8 \\ \hline \end{array}$$

Seperti yang dinyatakan di atas, setiap kali hasil tambah melebihi 7, satu pembawa akan terhasil. Lebih daripada satu pembawa mungkin terhasil jika tiga atau lebih nombor ditambah seperti dalam contoh di sebelah:

$$\begin{array}{r}
 7_8 \\
 7_8 \\
 + 7_8 \\
 \hline
 \end{array}$$

Hasil tambah digit pertama dan digit kedua adalah 6_8 dengan satu pembawa. Hasil tambah 6_8 dan digit ketiga ialah 5_8 dengan satu pembawa.



2.3.2.4 Penolakan nombor oktal

Penolakan nombor oktal adalah sama dengan penolakan nombor dalam sistem nombor yang lain. Satu-satunya variasi adalah dalam kuantiti pinjaman. Dalam sistem nombor desimal, kita perlu meminjam satu kumpulan 10_{10} . Dalam sistem binari, kita meminjam satu kumpulan 2_{10} . Dalam sistem oktal, kita akan meminjam satu kumpulan 8_{10} .

Pertimbangkan penolakan 1 daripada 10 dalam sistem nombor desimal, binari dan oktal.

Desimal

Binari

Oktal

$$\begin{array}{r} 10_{10} \\ - 1_{10} \\ \hline 9_{10} \end{array}$$

$$\begin{array}{r} 10_2 \\ - 1_2 \\ \hline 1_2 \end{array}$$

$$\begin{array}{r} 10_8 \\ - 1_8 \\ \hline 7_8 \end{array}$$

Dalam setiap contoh, kita tidak boleh tolak 1 daripada 0 dan mempunyai perbezaan positif. Kita perlu menggunakan satu pinjaman dari kolum sebelah.

Contoh:

Desimal

Binari

Oktal

$$\begin{array}{r} 10 \\ 10_{10} \\ - 1_{10} \\ \hline 9_{10} \end{array}$$

$$\begin{array}{r} 2 \\ 10_2 \\ - 1_2 \\ \hline 1_2 \end{array}$$

$$\begin{array}{r} 8 \\ 10_8 \\ - 1_8 \\ \hline 7_8 \end{array}$$

Pinjaman

Bila pinjaman digunakan, kolum yang dipinjam berkurang satu dan jumlah pinjaman ditambah ke kolum sebelah kanan.

Contoh berikut menunjukkan prosedur ini.

$$\begin{array}{r} 2 \ 10 \\ 3 \ 4_{10} \\ - 9_{10} \\ \hline 25_{10} \end{array}$$

$$\begin{array}{r} 3 \ 10 \\ 4 \ 6_8 \\ - 7_8 \\ \hline 3 \ 7_8 \end{array}$$

1.3.3 **Contoh 3:** Dalam contoh oktal, 7_8 tidak boleh ditolak daripada 6_8 , jadi pinjaman daripada 4 dilakukan. Kurangkan 1 daripada 4 dan tambah 10_8 (pinjaman) kepada 6_8 . Dengan menolak 7_8 daripada 16_8 , perbezaannya adalah 7_8 . Tuliskan nombor ini dan turunkan digit 3.

Binary	Decimal	Octal
000	0	0
001	1	1
010	2	2
011	3	3
100	4	4
101	5	5
110	6	6
111	7	7
1000	8	10
1001	9	11
1010	10	12
1011	11	13
1100	12	14
1101	13	15

2.3.3 Sistem nombor heksadesimal

[2] Disebabkan 16 nombor berbeza perlu diwakili, digit 0 hingga 9 tidak mencukupi maka abjad A, B, C, D, E dan F juga digunakan.

Jadual 2.3: 16 nombor binari, heksadesimal dan desimal yang sepadan.

Binari	Heksadesimal	Desimal
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7
1000	8	8
1001	9	9
1010	A	10
1011	B	11
1100	C	12
1101	D	13
1110	E	14
1111	F	15

2.3.3.1 Penukaran nombor desimal ke nombor heksadesimal

Untuk menukarkan nombor desimal ke heksadesimal, lebih mudah jika kita menukarkan dahulu nombor desimal ke binari diikuti binari ke heksadesimal. Untuk menukarkan nombor binari ke heksadesimal, digit-digit nombor binari dibahagikan kepada empat bahagian bermula dari kanan. Kemudian darabkan setiap digit dalam setiap bahagian bermula dari kanan dengan 2^0 , 2^1 , 2^2 , 2^3 dan seterusnya jumlahkan hasil darab setiap digit dalam setiap bahagian.

Contoh:

$$50_{10} = 30_{16}$$

2	50
2	25 ... 0
2	12 ... 0
2	6 ... 0
2	3 ... 0
	1 ... 1

11	0000
3	0

2.3.3.2 Penukaran nombor heksadesimal ke nombor desimal

Penukaran nombor heksadesimal kepada nombor desimal boleh dilakukan dengan menukar setiap digit nombor heksadesimal ke nombor desimal yang sepadan. Kemudian setiap digit didarab dengan 16^{n-1} yang mana n adalah posisi digit nombor heksadesimal tersebut. Selepas itu, hasil darab dijumlahkan.

Contoh:

$$\begin{aligned} AB_{16} &= 10_{10} \times 16^2_{10} + 11_{10} \times 16_{10} + 6_{10} \\ &= 10_{10} \times 256_{10} + 176_{10} + 6_{10} \\ &= 2560_{10} + 176_{10} + 6_{10} \\ &= 2742_{10} \end{aligned}$$

2.3.3.3 Penambahan nombor heksadesimal

Penambahan nombor heksadesimal tidak berbeza daripada penambahan dalam sistem nombor yang lain. Selagi hasil tambah dua nombor adalah 15_{10} atau kurang, hanya satu simbol digunakan untuk hasil tambah. Pembawa akan terhasil apabila hasil tambah dua nombor adalah 16_{10} atau lebih, seperti dalam contoh berikut.

8_{16}	A_{16}	D_{16}	$\overset{1}{7} \overset{1}{8} 4_{16}$
$+ 8_{16}$	$+ D_{16}$	$+ 9_{16}$	$+ BDA_{16}$
$\hline 10_{16}$	$\hline 17_{16}$	$\hline 16_{16}$	$\hline 13 \ 5 \ E_{16}$

Di sini, hasil tambah 4 dan A ialah E. Hasil tambah 8 dan D ialah 15₁₆; 5 ditulis dan 1 dibawa ke atas. Tambah pembawa dengan 7 dan B; tulis 3 dan bawa 1 ke atas. Disebabkan hanya pembawa yang tinggal, maka turunkannya ke bawah untuk selesaikan masalah tersebut.

2.3.3.4 Penolakan nombor heksadesimal

Penolakan nombor heksadesimal kelihatan lebih susah daripada sebenar. Penolakan adalah seperti nombor yang lain. Cuma simbol dan jumlah pinjaman berbeza.

$$\begin{array}{r} ABC_{16} \\ + 642_{16} \\ \hline 47A_{16} \end{array} \qquad \begin{array}{r} 7E5E_{16} \\ - 471_{16} \\ \hline 3744_{16} \end{array}$$

2.4 Web Calculator

Web calculator merupakan kalkulator yang boleh didapati secara atas talian yang mana pengguna perlu masuk ke halaman-halaman tertentu untuk mendapatkan perkhidmatan ini. Terdapat pelbagai jenis *web calculator* yang terdapat di dalam Internet. Mereka menyediakan perkhidmatan dalam berbagai-bagai jenis pengiraan contohnya pengiraan dalam bidang kejuruteraan, kewangan, kesihatan dan lain-lain lagi.

2.5 Kalkulator Binari, Oktal, Desimal dan Heksadesimal

Namun begitu, sistem yang akan dibangunkan adalah *web calculator* untuk pengiraan nombor binari, oktal, desimal dan heksadesimal. Sistem ini mampu menjalankan pengiraan-pengiraan asas iaitu penambahan, penolakan, pembahagian dan juga pendaraban. Selain itu terdapat fungsi tambahan untuk nombor binari iaitu fungsi AND, OR, NOT dan XOR. Setelah kajian dilakukan, terdapat beberapa *web calculator* berasaskan nombor-nombor ini yang wujud secara *stand-alone* dan juga di dalam Internet. Antaranya ialah:-

2.5.1 Stand-alone

a) Windows

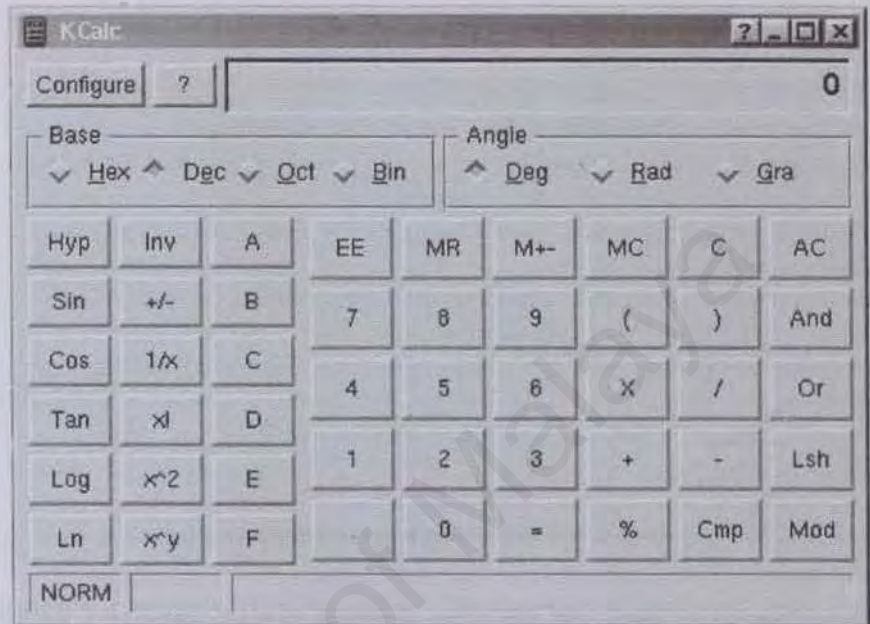


Rajah 2.2: Kalkulator di dalam sistem pengoperasian Windows

Kalkulator dari pakej pengoperasian Windows ini menyediakan kemudahan pengiraan bagi sistem nombor desimal, binari, oktal dan heksadesimal. Pengguna hanya perlu memilih sistem nombor yang dikehendaki dan nombor-nombor dan operasi-operasi yang berkaitan dengan sistem nombor tersebut akan dipaparkan. Nombor-nombor dan operasi-operasi yang tidak berkaitan dengan sistem nombor yang dipilih akan dikelamkan. Ini akan memudahkan pengguna menggunakan kalkulator ini tanpa sebarang kekeliruan. Selain itu, susunan butang-butang dan komponen-komponen lain juga disusun dengan teratur. Ini menampakkan lagi kekemasan kalkulator ini dan menjadikan ia mudah untuk digunakan. Menu bantuan (*help*) juga disediakan

bagi membantu pengguna menggunakan kalkulator ini dengan lebih baik.

b) Linux



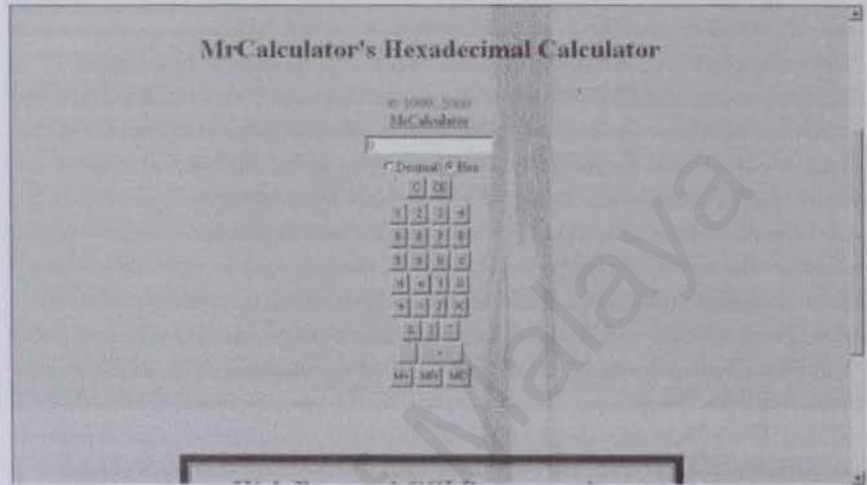
Rajah 2.3: Kalkulator di dalam sistem pengoperasian Linux

Kalkulator di dalam sistem pengoperasian Linux ini adalah sama seperti kalkulator di dalam sistem pengoperasian Windows iaitu menyediakan kemudahan untuk pengiraan semua sistem nombor. Namun begitu, terdapat sedikit perbezaan di dalam susun atur butang-butang tetapi ia masih lagi teratur. Nombor-nombor dan operasi-operasi yang tidak berkaitan dengan sistem nombor yang dipilih juga akan dikelamkan bagi memudahkan dan mengelakkan kekeliruan pengguna. Di samping itu, menu bantuan

(help) juga disediakan bagi membantu pengguna menggunakan kalkulator ini.

2.5.2 Berasaskan web (*web-based*)

a) MrCalculator's Heksadesimal Calculator

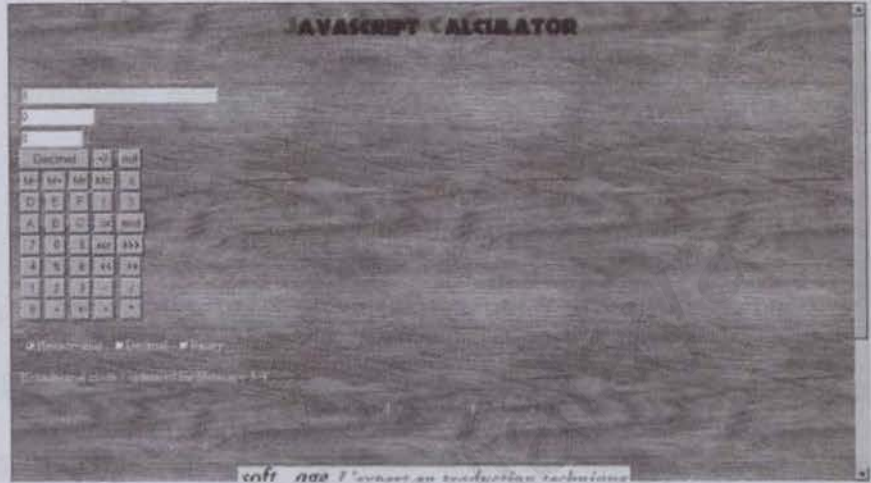


Rajah 2.4: Kalkulator berasaskan web MrCalculator's Hexadecimal Calculator

Kalkulator ini hanya menyediakan kemudahan untuk pengiraan nombor desimal dan heksadesimal sahaja. Dua butang radio diletakkan di bahagian atas kalkulator untuk memudahkan pengguna memilih sistem nombor yang dikehendaki. Kelemahan kalkulator ini adalah butang-butangnya tidak tersusun dengan rapi. Ini menyukarkan pengguna untuk memilih butang-butang tersebut. Selain daripada itu, apabila pengguna memilih opsi sistem nombor desimal, huruf-huruf a, b, c, d, e dan f tidak

dipadam atau dikaburkan. Ini mengelirukan pengguna yang tidak mahir dalam penggunaan sistem nombor ini.

b) Javascript Calculator



Rajah 2.5: Kalkulator berasaskan web Javascript Calculator

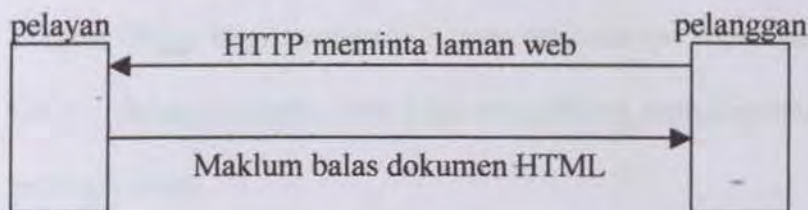
Kalkulator ini pula membolehkan pengguna memilih sama ada pengiraan untuk nombor heksadesimal, desimal atau binari. Walau bagaimanapun pengiraan untuk nombor oktal tidak disediakan. Di ruang atas kalkulator ini, tiga ruang output diletakkan. Ini boleh mengelirukan pengguna untuk memastikan output yang manakah sebenarnya. Tiga ruang output diletakkan di atas bagi mewakili ketiga-tiga sistem nombor tersebut iaitu, binari di bahagian atas, diikuti desimal dan heksadesimal. Apabila nombor dimasukkan (contohnya nombor binari), kalkulator akan menukarkan nombor tersebut ke dalam nombor desimal dan heksadesimal yang sepadan. Ini sudah tentu mengelirukan

pengguna yang pertama kali menggunakan kalkulator ini untuk mengecam output bagi setiap sistem nombor. Walau bagaimanapun, butang-butang disusun dengan kemas dan butang yang tidak berkaitan dengan sistem nombor tersebut akan dipadamkan apabila sesuatu sistem nombor itu dipilih. Ia membantu pengguna memilih nombor yang tepat setiap kali pengiraan dibuat.

2.6 Aplikasi Web

Mengikut sejarah, aplikasi web berasaskan kepada program Hyper Text Markup Language (HTML) dan Common Gateway Interface (CGI) pada pelayannya. Laman web boleh merangkumi kombinasi teks berformat, imej dan grafik, audio dan video. HTML juga membenarkan penghasilan borang asas yang mengandungi ruang teks, butang radio, kotak semakan, butang tekan dan kotak senarai. Pada kebiasaannya, pelayar (*browser*) dilayan sebagai pelanggan universal yang menghantar satu permintaan untuk laman web, menafsir dokumen HTML dan memaparkannya kepada pengguna. Pelayan web menerima permintaan melalui Hyper Text Transport Protocol (HTTP) dan memulangkan maklumat yang diperlukan di dalam format HTML yang boleh difahami oleh pelanggan.

Aplikasi web juga boleh wujud dari aplikasi pelanggan/pelayan tradisional tetapi memerlukan perancangan yang meluas dan ia bukanlah satu tugas yang mudah. Aplikasi web menggunakan *Internet protocol* seperti TCP/IP, HTTP dan HTML untuk paparan maklumat dan protokol rangkaian.



Rajah 2.6: Senibina aplikasi web

2.7 Kajian Bahasa Pengaturcaraan

Bahasa pengaturcaraan yang dikaji adalah Java dan C++. Kedua-dua bahasa ini dibuat perbandingan bagi memastikan bahasa pengaturcaraan yang sesuai digunakan untuk membangunkan sistem.

2.7.1 Java

[6] Java ialah satu bahasa pengaturcaraan komputer berorientasikan objek yang dibangunkan oleh Sun Microsystems dalam tahun 1991. Pada asalnya, Java dicipta untuk membangunkan perisian untuk produk pengguna seperti set televisyen, pembakar roti dan VCR. Untuk mencapai tujuan ini, pencipta Java mahu menjadikannya kecil, mudah, cekap dan mudahalih kepada satu julat besar peranti perkakasan. Atas sebab yang sama, selepas itu Java juga adalah satu bahasa yang sesuai untuk membangunkan perisian untuk pengagihan melalui World Wide Web (WWW). WWW adalah selepas semua koleksi sumber-sumber disimpan di dalam komputer dilarikan di atas pelbagai sistem pengoperasian.

Java adalah seperti C/C++. Sintaksnya hampir sama dengan C/C++. Walau bagaimanapun, Java menghindarkan beberapa kompleksiti C/C++. Sebagai contoh, Java tidak menyokong penuding atau perwarisan pelbagai kelas.

Java adalah bahasa pengaturcaraan kegunaan-am dan bahasa pengaturcaraan Internet. Program Java boleh dilarikan sebagai aplikasi stand-alone dengan sendirinya seperti program yang ditulis dalam bahasa Pascal atau C/C++. Tetapi program Java juga boleh dilarikan dalam bentuk applets. Applets adalah program yang dinamik dan interaktif; ia dilarikan di bawah satu pelayar yang menyokong Java seperti HotJava, Internet Explorer atau Netscape Communicator. Ini bermakna kod Java (dalam bentuk applets) oleh diagihkan melalui World Wide Web untuk dilarikan di atas pelbagai mesin dan platform sistem pengoperasian.

Ciri-ciri utama Java:

1. *Mudah dipelajari dan digunakan*

Disebabkan Java dicipta kecil dan mudah, maka program Java mudah disunting, dikompil, debug dan dilarikan.

2. *Platform yang bebas.*

Program Java boleh dikompil dan dilarikan di atas platform yang berbeza. Ini bermakna, kod tersebut tidak perlu dikompil semula apabila kod diagihkan ke platform yang berbeza. Kebebasan

platform boleh digunakan di kedua-dua sumber dan fail binari. Ini berlaku kerana fail sumber Java dicipta menggunakan satu pengedit teks seperti Notepad (di bawah Aksesori Windows). Selepas fail sumber dikompil, fail binari disimpan di dalam bentuk kod bait mesin-neutral. Kod bait adalah kod (arahan) yang seakan-akan kod mesin tetapi ia tidak spesifik kepada mesin tertentu. Kod bait mesin-neutral kemudian boleh dilaksanakan di atas mana-mana mesin yang mempunyai pentafasir Java (kod bait).

3. *Dikompil dan ditafsir*

Program Java pada mulanya dikompil ke dalam kod bait mesin-neutral (kod bait adalah set arahan yang seakan-akan kod mesin tetapi ia tidak spesifik kepada mana-mana satu pemproses). Kemudian, kod-kod bait ditafsir oleh satu penterjemah kod bait Java atau oleh pelayar yang menyokong Java seperti Internet Explorer atau Netscape Communicator. Modul (program) di dalam pelayar tersebut yang melaksanakan kod-kod bait dipanggil Java Virtual Machine (JVM). JVM menafsir applet (dalam bentuk kod-kod bait) dan memastikan ia berfungsi dengan baik di atas komputer.

4. *Berorientasikan objek.*

Pada keseluruhannya, Java adalah berorientasikan objek. Ini bermakna, banyak faedah dapat diperolehi apabila menggunakan teknologi berorientasikan objek seperti penggunaan semula perisian, kefleksibelan dan modulariti.

5. *Teguh*

Teguh merujuk kepada tahap sesuatu aplikasi boleh mengawal semua keadaan sama ada yang dijangka atau tidak. Lebih teguh sesuatu aplikasi itu, lebih kuat ia menentang serangan. Aplikasi yang kurang teguh selalunya senang dirempuh masuk. Sistem Java memeriksa setiap capaian memori dengan teliti untuk memastikan ia adalah sah. Dalam keadaan ini, ralat dihapuskan atau dikurangkan. Apabila ralat berlaku, Java membuang satu pengecualian. Ia mengesan ralat dan membenarkan program memulihkan dengan baik. Salah satu sebab kenapa program Java adalah teguh kerana ia tidak membenarkan bahagian ingatan komputer yang kritikal kepada sistem prestasi ditukar.

6. *Pautan dinamik*

Ciri ini juga dikenali sebagai *delayed binding*. Ia menawarkan lebih fleksibiliti. Di dalam pautan dinamik, modul dipautkan hanya apabila diperlukan. Modul yang tidak diperlukan atau

jarang diminta (cth: rutin pemegangan pengecualian) tidak dipautkan. Ini mengurangkan saiz program.

7. *Keselamatan*

Applet Java tidak boleh merempuh masuk komputer disebabkan ia tidak dibenarkan untuk mencapai fungsi sistem-kritikal. Ini kerana, kod-kod bait Java mengandungi maklumat jenis tambahan yang akan disemak dua kali untuk kesahihan. Penafsir Java memeriksa kod-kod bait sebelum ia dilaksanakan sebenarnya.

8. *Multithreading*

Ciri ini membenarkan beberapa *threads* dilarikan secara serentak di dalam program yang sama tanpa mengganggu antara satu sama lain. Ini bermakna, ia boleh beberapa applet boleh dilarikan secara serentak di dalam halaman yang sama.

2.7.2 C++

[5] Bahasa C++ direkacipta oleh Bjarne Stroustrup pada awal 1980an. Antara objektif Stroustrup semasa pembangunan bahasa C++ ialah untuk menjadikannya serasi dengan C dan untuk memasukkan binaan-binaan dari Simula67 ke dalam bahasa C.

[1] C++ adalah satu bahasa pengaturcaraan yang diwarisi dari C. Ia adalah superset C yang ditambah dengan pemeriksaan jenis, *operator overloading*, jenis data abstrak dan kelas-kelas kepada bahasa asal. Ia

menggabungkan pengaturcaraan berorientasikan objek dengan kecekapan dan notasi C.

[5] C++ adalah bahasa mudah alih dan terdapat pelbagai jenis pengkompil C++ yang telah dibangunkan. Ia juga mempunyai ciri-ciri yang lebih baik dari C terutama ianya lebih berjenis kuat (*strong typing*), lebih selamat dan ditambah pula dengan ciri-ciri berorientasikan objek.

C++ ialah bahasa yang menggabungkan pengaturcaraan aras bawah dan aras tinggi. Oleh itu, pengguna boleh menulis aturcara pada aras yang ia perlukan bagi sesuatu penyelesaian masalah. C++ juga tidak memberi penekanan kepada prapemproses tetapi ianya mempunyai satu katakunci "*inline*" yang sangat berguna bagi memohon pengkompil menghasilkan kod *inline* bagi sesuatu fungsi.

Pengabstrakan data dapat dilaksanakan dalam C++ melalui mekanisme "*class*". *Class* membenarkan pengaturcara mengawal capaian terhadap sesuatu kod perlaksanaan. C++ juga memastikan penghantaran parameter disemak melalui prototaip fungsinya. Fungsi yang menggunakan pelbagai jenis dan pelbagai jumlah hujah juga dibenarkan bagi memudahkan penggunaan fungsi.

2.8 Kajian Metodologi

Terdapat pelbagai metodologi yang boleh digunakan untuk memodelkan proses perisian. [4] Dengan membina satu model proses dan membincangkan subproses-subprosesnya, ini dapat membantu pasukan pembangun memahami jurang antara apa yang akan terjadi dan apakah jurang tersebut. Berikut adalah beberapa contoh model proses:

2.8.1 Model Air Terjun

[4] Model ini dipanggil model air terjun kerana ia digambarkan seperti air terjun yang turun dari satu peringkat ke peringkat lain. Satu peringkat pembangunan perlu disiapkan sebelum peringkat selepasnya bermula. Model ini menunjukkan satu tahap pandangan yang sangat tinggi tentang apa yang berlaku semasa pembangunan dan ia menyarankan kepada pembangun turutan aktiviti yang perlu mereka jangka untuk ditempuhi. Model ini telah digunakan untuk menetapkan aktiviti-aktiviti pembangunan perisian dalam pelbagai konteks. Setiap aktiviti proses terdapat *milestones* dan *deliverables*, maka pengurus-pengurus projek boleh menggunakan model ini untuk menilai masa yang tinggal untuk menyiapkan projek.

2.8.2 Model Air Terjun dengan Prototaip

Model proses ini adalah sama dengan model air terjun Cuma ia hanya ditambah dengan prototaip. [4] Pemprototaipan adalah satu subproses; prototaip adalah satu produk yang telah dibangunkan

sebahagian yang membolehkan pelanggan-pelanggan dan pembangun-pembangun mengenalpasti beberapa aspek sistem yang dikehendaki dan memutuskan jika ia sesuai untuk produk akhir.

2.8.3 Model V

[4] Model V adalah satu variasi model air terjun yang mendemonstrasikan bagaimana aktiviti-aktiviti pengujian berkait dengan analisis dan rekabentuk. Model V mencadangkan pengujian unit dan integrasi juga digunakan untuk mengesahkan rekabentuk program. Pautan model sebelah kiri dengan sebelah kanan V menerangkan bahawa jika masalah didapati semasa pengesahan dan penilaian, kemudian bahagian kiri V boleh dilaksanakan semula untuk memperbetul dan memperbaiki rekabentuk keperluan dan kod sebelum langkah pengujian di sebelah kanan dilakukan semula (*reenacted*). Model V menjadikan beberapa iterasian dan *rework* lebih jelas dan menjalankan semula yang tersembunyi di dalam gambaran air terjun. Fokus model V adalah aktiviti dan pembetulan.

2.8.4 Model Prototaip

[4] Model prototaip membenarkan semua atau sebahagian sistem dibina dengan cepat untuk memahami atau menerangkan isu-isu. Ia mempunyai objektif yang sama dengan prototaip kejuruteraan, yang mana keperluan-keperluan atau rekabentuk memerlukan pemeriksaan yang berulang untuk memastikan pembangun, pengguna dan pelanggan

mempunyai pemahaman yang sama apa yang diperlukan dan apa yang dicadangkan.

2.8.5 Fasa Pembangunan: Penokokan dan Iterasian

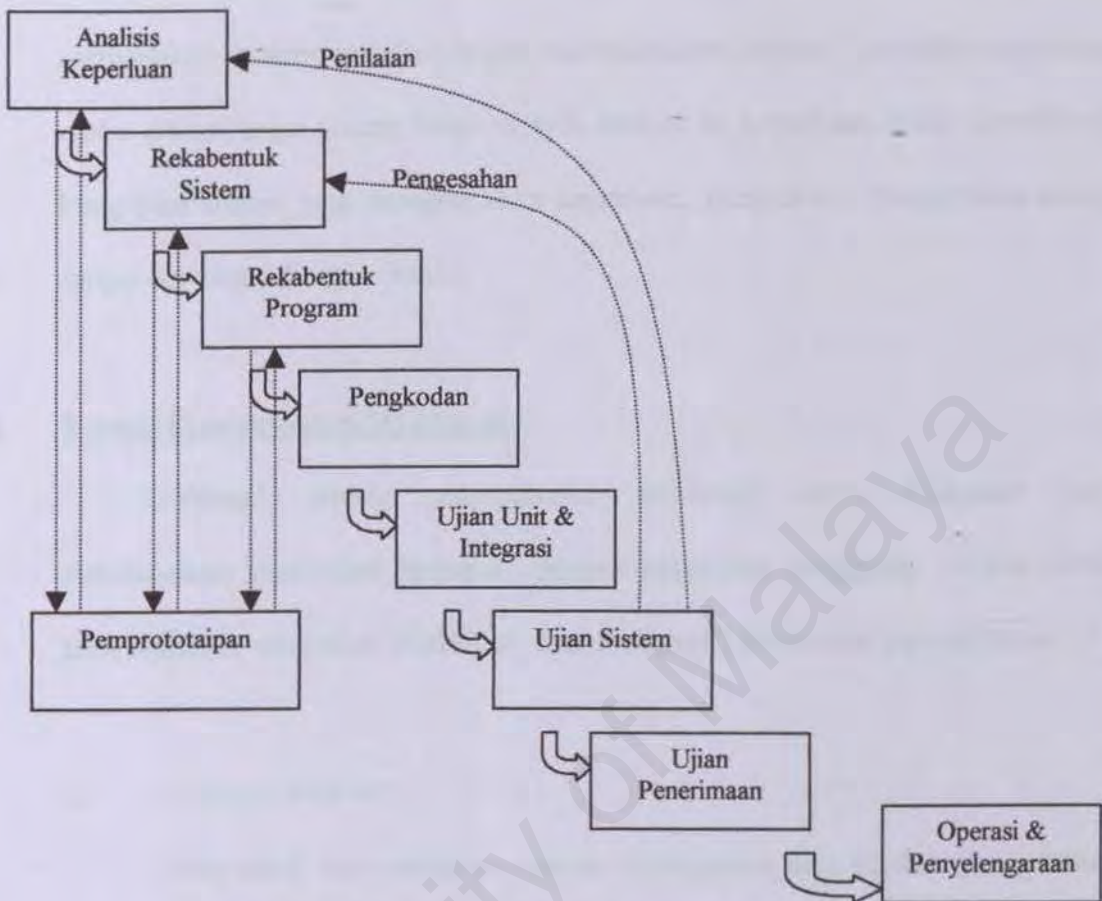
[4] Sistem ini direka untuk membolehkan penghantaran dalam sebahagian, membolehkan pengguna-pengguna mempunyai beberapa kefungsian sementara yang selebihnya sedang dibangunkan. Kebiasaannya, dua sistem berfungsi dalam jujukan: sistem produksi dan sistem pembangunan. Sistem operasian atau produksi-produksi adalah yang digunakan oleh pelanggan dan pengguna pada masa sekarang, sistem pembangunan adalah versi berikut yang sedang disediakan untuk menggantikan sistem produksi sekarang. Terdapat banyak cara untuk pembangun-pembangun menentukan bagaimana untuk menyelaraskan pembangunan kepada 'release'. Dua kaedah yang paling popular adalah pembangunan penokokan dan pembangunan iterasian. Dalam pembangunan penokokan, sistem yang dispesifikkan dalam dokumen keperluan dibahagikan kepada subsistem yang kecil dan berfungsi dan fungsian ditambah dengan setiap release baru. Pembangunan iterasi menghasilkan satu sistem penuh pada permulaan dan kemudian menukar fungsian setiap subsistem dengan setiap release baru.

2.9 Rumusan

Secara keseluruhan, bab ini mengkaji permasalahan yang dijalankan sebelum projek dapat dilaksanakan. Kajian-kajian ini meliputi kajian terhadap sistem terdahulu, kajian terhadap bahasa pengaturcaraan dan kajian metodologi untuk dipilih mengikut kesesuaian pembangunan sistem ini.



3.1 Pemilihan Metodologi



Rajah 3.1: Model Air Terjun dengan Prototaip

Di dalam menghasilkan *web calculator* ini, model air terjun dengan prototaip dipilih sebagai model proses perisian. Model ini dipilih kerana [4] prototaip yang dibangunkan membolehkan pelanggan dan pembangun memeriksa beberapa aspek sistem yang dicadangkan dan menentukan jika ia sesuai untuk produk akhir. Di samping itu, ia juga menjimatkan kos kerana rujukan dilakukan pada peringkat keperluan. Melalui rekabentuk prototaip, ia dapat membantu pembangun-pembangun menilai strategi-strategi rekabentuk alternatif dan menentukan rekabentuk yang terbaik untuk sesuatu projek.

Kekusutan utama dalam keperluan-keperluan dikenalpasti dan dibetulkan sebelum keperluan-keperluan dinilai semasa pengujian sistem; penilai memastikan sistem tersebut telah melaksanakan semua keperluan-keperluan, maka setiap fungsi sistem boleh dijejak semula ke keperluan dalam spesifikasi. Pengujian sistem juga mengesahkan keperluan, pengesahan memastikan setiap fungsi berfungsi dengan betul.

3.2 Teknik Pengumpulan Maklumat

Pelbagai teknik pengumpulan maklumat telah dilakukan bagi mendapatkan maklumat berkaitan dengan keperluan pengguna. Teknik-teknik pengumpulan yang telah dilakukan ialah melayari Internet dan penyelidikan.

a) *Melayari Internet*

Pada masa kini, melayari internet merupakan satu kaedah yang cekap dan cepat dalam mengumpulkan maklumat. Terdapat banyak laman web yang menyediakan maklumat yang berguna dan diperlukan di dalam sistem yang akan dibangunkan. Beberapa laman web menyediakan kalkulator seperti sistem yang akan dibangunkan yang sangat berguna untuk dijadikan sebagai garis panduan dalam menentukan keperluan sistem.

b) *Penyelidikan*

Penyelidikan melibatkan pengkajian buku-buku yang mengandungi maklumat yang berkaitan. Maklumat-maklumat ini dikumpul dan digabungkan bagi membentuk keperluan-keperluan sistem.

3.3 **Rumusan**

Di dalam bab ini, pemilihan metodologi dilakukan dan teknik yang digunakan bagi menyelesaikan masalah projek dikemukakan. Metodologi yang dipilih adalah model air terjun dengan prototaip dan teknik pengumpulan maklumat yang dilakukan adalah melayari internet dan penyelidikan.

ANALISA SISTEM

4.1 Keperluan Fungsian

[3] Keperluan fungsian menerangkan interaksi antara sistem dan persekitarannya. Keperluan fungsian di dalam sistem ini ialah:

- a) Melakukan operasi penambahan, penolakan, pendaraban, pembahagian, AND, OR, NOT dan XOR bagi sistem nombor binari, oktal, dan heksadesimal.
- b) Melakukan operasi penambahan, penolakan, pendaraban dan pembahagian bagi sistem nombor desimal.
- c) Menu bantuan (*help*)
- d) Butang pemilihan sistem nombor.

4.2 Keperluan Bukan Fungsian

[3] Keperluan bukan fungsian ialah deskripsi bagi ciri-ciri yang menyempurnakan lagi sesuatu sistem maklumat dan juga kekangan-kekangan yang menghadkan sempadan atau skop. Di samping keperluan fungsian, lain-lain keperluan bukan fungsian adalah sesuai untuk ditambah. Keperluan-keperluan ini terdiri daripada:-

- a) *Kebolehpercayaan*

Keperluan ini amat penting bagi memastikan kepercayaan pengguna terhadap sistem yang akan dibangunkan. Kebolehpercayaan ini merangkumi hasil kiraan yang dipaparkan adalah betul dan tepat.

b) *Masa tindak balas*

Masa tindak balas bagi sistem ini mestilah cepat. Masa tindak balas yang dianggarkan adalah kurang daripada dua saat. Sebaik sahaja pengguna menekan butang '=', sistem akan memaparkan outputnya.

c) *Mesra pengguna*

Sistem mestilah berada di dalam persekitaran yang mesra pengguna dan boleh digunakan oleh sesiapa sahaja yang tidak mempunyai pengetahuan tentangnya.

d) *Antaramuka yang menarik*

Sistem mestilah mempunyai antaramuka yang menarik agar pengguna tidak jemu dan berminat untuk menggunakan lagi sistem yang dibangunkan.

4.3 Pemilihan Bahasa Pengaturcaraan

Bahasa pengaturcaraan yang dipilih bagi membangunkan sistem ini ialah Java. Bahasa ini dipilih kerana:

a) *Berorientasikan objek*

Platform Java menyokong pengaturcaraan berorientasikan objek. Ini menyebabkan Java adalah bahasa yang sesuai untuk membangunkan kalkulator kerana ia membantu dalam menguruskan kekompleksitian dan penggunaan semula kod. Maka masa pembangunan dapat dikurangkan.

b) *Teragih*

Web adalah medium revolusi untuk pengagihan maklumat. Kalkulator yang dibina menggunakan platform Java membenarkan pengguna yang mempunyai pelayar web mencapainya.

c) *Pelbagai platform*

Di dalam kes ini, kalkulator perlu membekalkan pelbagai *browser* dan platform pelanggan (*client*) yang mungkin. Java menyokong pelbagai platform. Dalam persekitaran Java, program Java dikompil kepada kod bait yang mana dilaksanakan oleh Java Virtual Machine (JVM). Oleh yang demikian, sebarang platform yang melaksanakan JVM mampu melarikan kod bait. Ini bermakna, pengaturcara boleh kompil kalkulator Java sekali dan ia akan dilarikan di atas sebarang komputer yang menyokong Java seperti Netscape Navigator. Kombinasi sokongan multiplatform dan pengagihan menyeluruh melalui web meningkatkan bilangan pengguna menggunakan kalkulator Java.

d) *Mudah*

Java dilihat sebagai bahasa yang mudah dalam beberapa sudut. Pertama, sintaks Java menyerupai C dan C++. Maka ia tidak terlalu susah kepada pengaturcara C dan C++ mempelajarinya. Kedua, Java menghapuskan ciri-ciri dari C dan C++ iaitu limpahan (*redundant*) atau penulisan kod yang tidak kemas. Maka Java adalah bahasa lebih kecil dan mudah berbanding C atau C++. Ketiga, Java menjadikan pengurusan ingatan

mudah melalui penghapusan penunding dan penggunaan pengumpulan sampah (*garbage collection*) untuk menebusguna ingatan yang tidak digunakan setelah diperuntukkan. Ini membolehkan kod pepijat dikurangkan. Platform Java yang mudah ini menjadikan kalkulator lebih mudah difahami dan dinyahpijat.

4.4 Keperluan Perkakasan

Keperluan perkakasan dibahagikan kepada dua kategori iaitu persekitaran pembangunan dan persekitaran masa-larian (pengguna).

a) Persekitaran Pembangunan

Keperluan-keperluan perkakasan minimum yang disyorkan untuk persekitaran pembangunan adalah seperti berikut:

- Komputer
- Pemproses Pentium 166Mhz atau setara.
- 64MB RAM.
- 3GB cakera keras.
- 28.8kbps modem.
- Papan kekunci dan tetikus sebagai peranti input.

b) Persekitaran Masa-larian

Keperluan-keperluan perkakasan minimum yang disyorkan untuk persekitaran masa-larian adalah seperti berikut:

- Komputer
- Pemproses Pentium 166MHz atau setara.
- 64MB RAM
- 4GB cakera keras
- 28.8kbps modem.
- Papan kekunci dan tetikus sebagai peranti input.

4.5 Keperluan Perisian

a) Persekitaran pembangunan

Keperluan-keperluan perisian minimum yang disyorkan untuk pembangunan sistem adalah seperti berikut:

- Java sebagai bahasa pengaturcaraan yang digunakan.
- Windows 95 atau setara.
- Internet Explorer versi 4.0 atau setara.

b) Persekitaran masa-larian

Keperluan-keperluan perisian minimum yang disyorkan untuk persekitaran masa-larian adalah seperti berikut:

- Java sebagai bahasa pengaturcaraan yang digunakan.
- Windows '95 atau setara.
- Internet Explorer 4.0 atau setara.

4.6 Rumusan

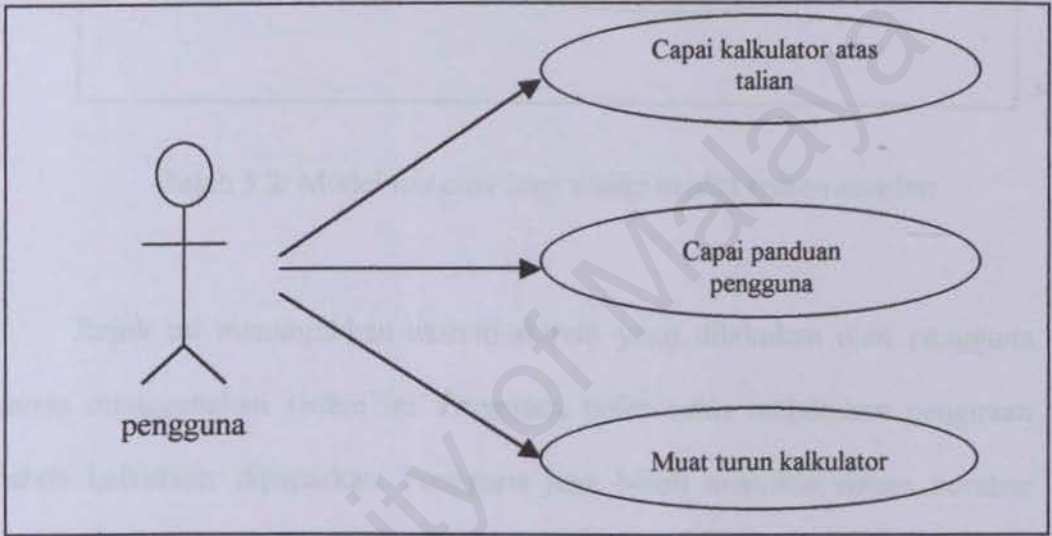
Bab ini menghuraikan keperluan-keperluan yang diperlukan dalam membangunkan sistem ini iaitu keperluan fungsian, keperluan bukan fungsian, keperluan perkakasan dan keperluan perisian. Di sini juga pemilihan bahasa pengaturcaraan dibuat. Bahasa pengaturcaraan yang dipilih adalah Java kerana ia adalah bahasa yang berorientasikan objek, teragih, mudah dan pelbagai platform.

REKABENTUK SISTEM

5.1 Pendahuluan

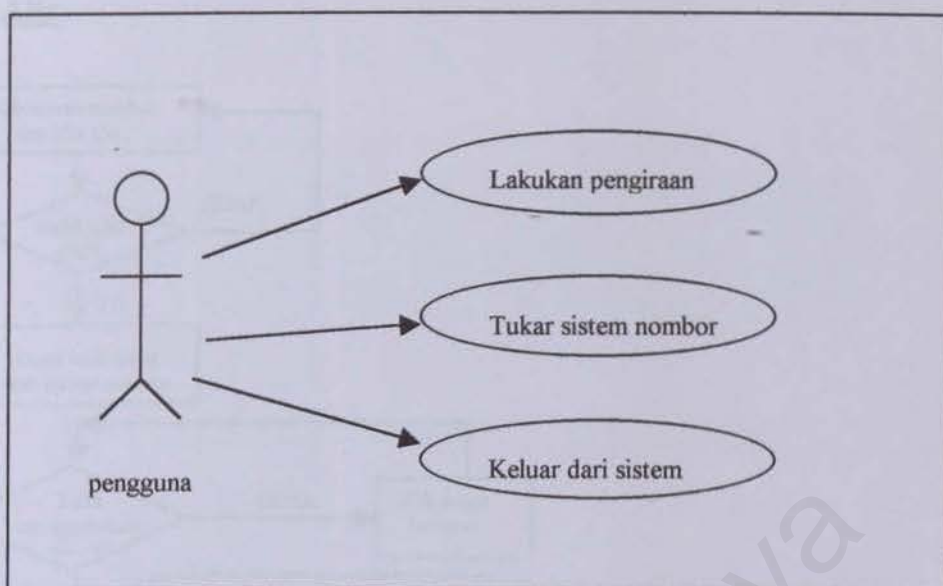
Bab ini menghuraikan tentang proses pencantuman kesemua bahagian-bahagian kalkulator kepada sebuah sistem. Ianya melibatkan modul-modul dan carta alir yang menerangkan tentang proses capaian kalkulator tersebut.

5.2 Model Use Case



Rajah 5.1: Model *use case* bagi keseluruhan sistem

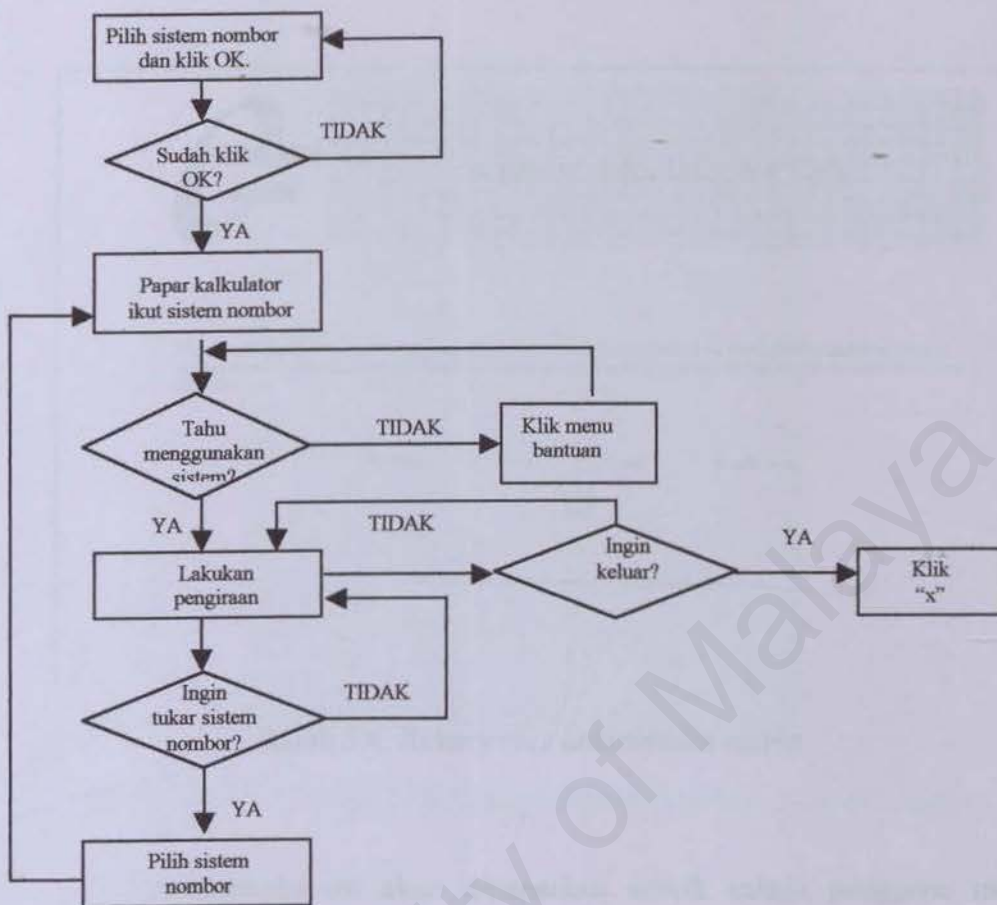
Rajah di atas menunjukkan rajah *use case* bagi sistem yang dibangunkan. Pengguna dapat mencapai kalkulator atas talian, panduan pengguna atau memuat turun kalkulator.



Rajah 5.2: Model *use case* bagi setiap modul sistem nombor

Rajah ini menunjukkan aktiviti-aktiviti yang dilakukan oleh pengguna semasa menggunakan sistem ini. Pengguna boleh terus melakukan pengiraan apabila kalkulator dipaparkan. Pengguna juga boleh menukar sistem nombor yang lain dan melihat menu bantuan (*help*) bagi membantu mereka dalam menggunakan sistem.

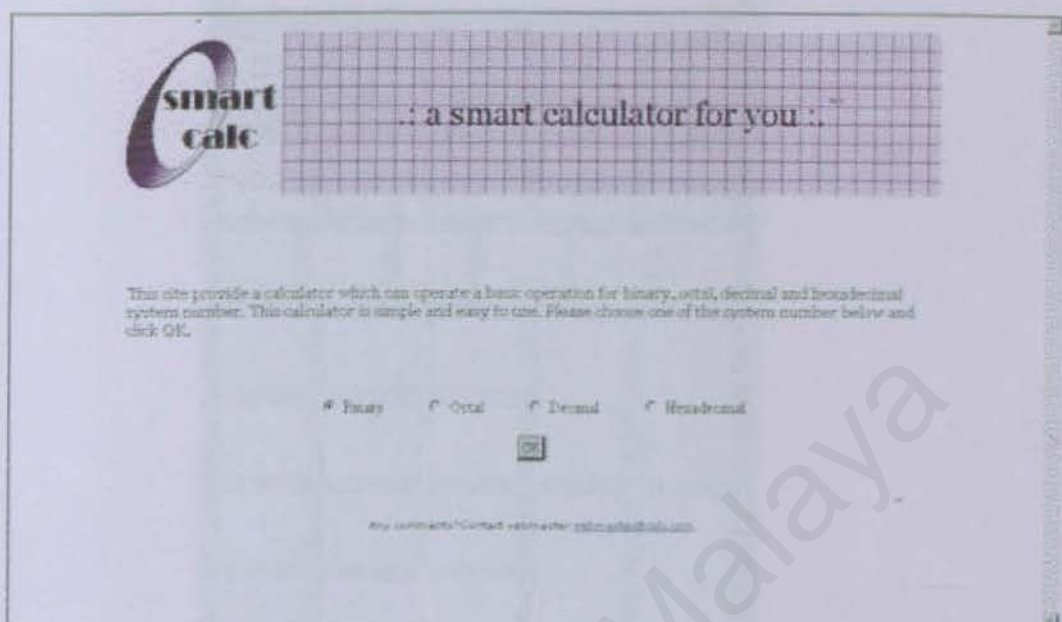
5.3 Carta Alir



Rajah 5.3: Carta alir bagi keseluruhan sistem

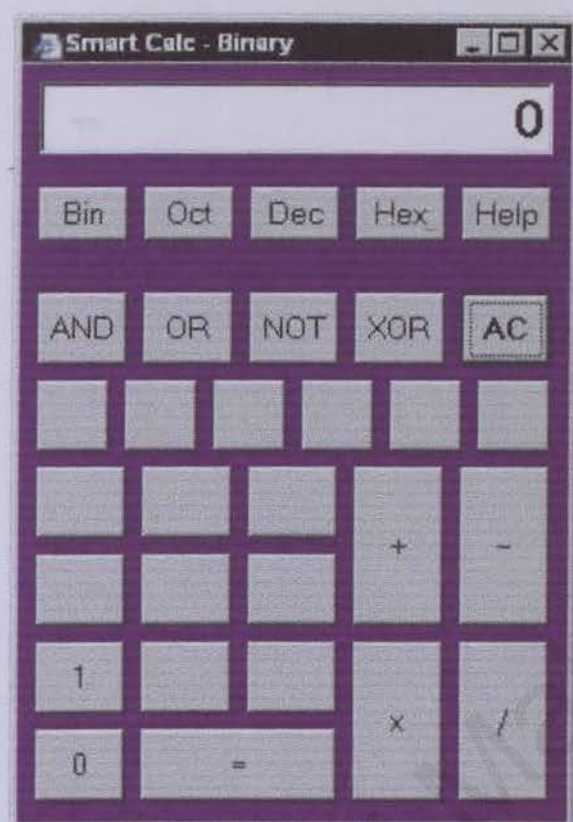
Rajah ini menerangkan aktiviti-aktiviti yang berlaku ketika pengguna mencapai sistem ini. Aktiviti-aktiviti yang terlibat ialah pemilihan sistem nombor, capaian menu bantuan dan sebagainya. Pada keseluruhannya, sistem ini amat mudah dan tidak sukar bagi pengguna yang kurang mahir dan pertama kali menggunakan sistem mencapainya.

5.4 Rekabentuk Antaramuka



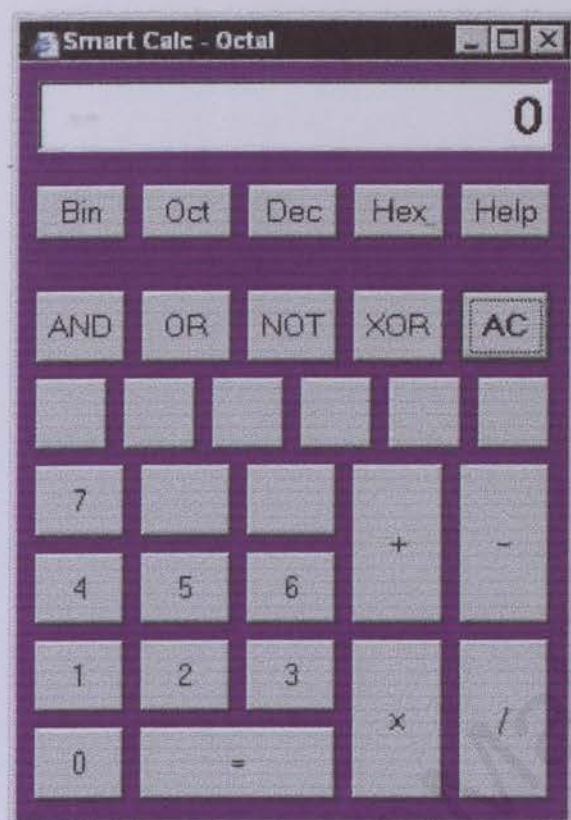
Rajah 5.4: Rekabentuk antaramuka utama

Antaramuka ini akan dipaparkan sebaik sahaja pengguna memasuki laman web ini. Di dalam laman ini terdapat empat butang radio bagi mewakili empat sistem nombor. Pengguna perlu memilih kalkulator yang dikehendaki berdasarkan sistem nombor tersebut dan klik 'OK'. Kemudian satu tettingkap kalkulator yang dipilih akan dipaparkan.



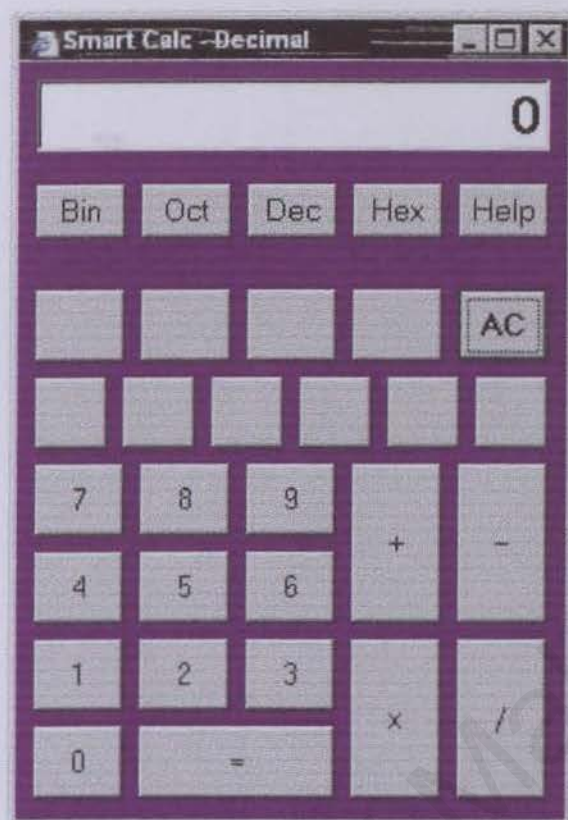
Rajah 5.5: Rekabentuk antaramuka kalkulator sistem nombor binari

Antaramuka ini akan dipaparkan sekiranya pengguna memilih kalkulator bersistem nombor binari. Kalkulator ini akan memaparkan jenis kalkulator yang sedang dipaparkan pada *title bar*. Semua operasi dan nombor yang melibatkan sistem nombor ini akan dipaparkan dan nombor yang tidak berkaitan akan dipadamkan. Terdapat juga butang pemilihan sistem nombor dan butang *help* di bawah ruang output.



Rajah 5.6: Rekabentuk antaramuka kalkulator sistem nombor oktal

Antaramuka kalkulator bagi sistem nombor oktal ini adalah sama sahaja dengan kalkulator sistem nombor binari. Perbezaannya di sini ialah ia memaparkan nombor dan operasi yang melibatkan sistem nombor oktal sahaja. Nombor dan operasi yang tidak berkaitan dengan sistem nombor akan dipadamkan.



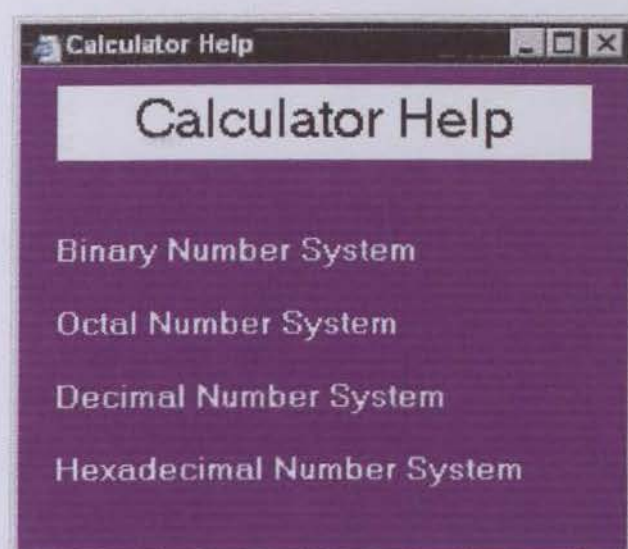
Rajah 5.7: Rekabentuk antaramuka kalkulator sistem nombor desimal

Antaramuka kalkulator bagi sistem nombor desimal ini juga adalah sama. Ia memaparkan nombor dan operasi yang melibatkan sistem nombor desimal sahaja. Nombor dan operasi yang tidak berkaitan dengan sistem nombor akan dipadamkan.



Rajah 5.8: Rekabentuk antaramuka kalkulator sistem nombor heksadesimal

Antaramuka kalkulator sistem nombor heksadesimal ini juga sama seperti kalkulator sistem nombor binari, oktal dan desimal. Keseragaman antaramuka ini membolehkan pengguna cepat memahami sistem dan mudah mempelajarinya. Perbezaannya hanya dari segi nombor. Hanya nombor yang berkaitan dengan sistem nombor sahaja akan dipaparkan manakala nombor yang tidak berkaitan akan dipadamkan. Ini akan memudahkan pengguna selain tidak mengelirukan.



Rajah 5.9: Rekabentuk antaramuka tetingkap bantuan

Tetingkap ini akan dipaparkan sebaik sahaja pengguna klik pada butang *help* pada tetingkap kalkulator. Terdapat empat pilihan sistem nombor. Pengguna hanya perlu klik pada sistem nombor yang diinginkan dan bantuan bagi sistem nombor yang dipilih akan dipaparkan.

PEMBANGUNAN SISTEM

6.1 Pengenalan

Pelaksanaan sistem adalah pembinaan sistem baru dan penghantaran sistem tersebut untuk pengeluaran. Ia melibatkan translasi perwakilan perisian yang dihasilkan oleh fasa rekabentuk ke dalam bentuk yang boleh dibaca oleh komputer. Fasa ini melibatkan ubahsuaian daripada rekabentuk yang lepas.

6.2 Dasar Pelaksanaan

Dasar berikut diikuti semasa fasa pelaksanaan *Web Calculator*:-

- **Pengasingan tugas** – Semua proses-proses pembangunan web melibatkan pengasingan proses-proses pembangunan web maka keputusan tentang struktur HTML yang spesifik adalah dibenarkan dibuat “*just in time*”. Semasa fasa pelaksanaan keputusan tentang web dibuat berdasarkan kepada toleran dan arahan yang didapati.
- **Kerja berterusan** – Pelaksanaan sistem web adalah proses berterusan. Maka, prosedur pelaksanaan web perlu direkabentuk dengan orientasi proses, membenarkan replikasi data, peningkatan dan kebolehpercayaan pengurusan fail dan teknik pengkodan.

6.3 Persekitaran Pembangunan

Selalunya, di dalam projek pemprototaipan perisian, analisis keperluan, rekabentuk sistem dan fasa-fasa pembangunan tidak mempunyai sempadan yang jelas. Setiap fasa cenderung untuk terlibat antara satu sama lain. Pembangunan sistem adalah satu proses, semasa menukarkan keperluan sistem dan rekabentuk

kepada kod program melibatkan sedikit perubahan dalam rekabentuk yang sebelumnya.

Pembangunan sistem terjemahkan kepada perwakilan lengkap perisian kepada bahasa pengaturcaraan yang sebenar. Proses penterjemahan berterusan apabila satu komputer menerima kod sumber sebagai satu input dan menghasilkan kod mesin. Di dalam melaksanakan proses tersebut, peralatan dan bahasa yang sesuai diperlukan untuk mengkodkan program tersebut. Beberapa peralatan perisian dipilih di dalam pembangunan *Web Calculator*.

6.3.1 Spesifikasi Perkakasan

Komponen spesifikasi perkakasan digunakan di dalam membangunkan atau melaksanakan sistem ini:-

Jadual 6.1: Spesifikasi Perkakasan

Komponen Perkakasan	Spesifikasi
Central Processing Unit (CPU)	Intel Pentium III 1.0GHz
Random Access Memory (RAM)	128MB
Cakera Keras	20 GB
Pemacu Disket	1.44MB
Monitor	15" SVGA
Papan Kekunci	
Tetikus	

6.3.2 Peralatan Perisian

- **Peralatan perisian untuk rekabentuk dan penulisan laporan**

Proses rekabentuk termasuk melukis carta struktur, rajah aliran data dan lain-lain lukisan yang membentuk asas pembangunan perisian. Tujuan rekabentuk logikal berasaskan grafik adalah untuk menyediakan pandangan menyeluruh sistem dan hubungan antara modul. Peralatan yang digunakan adalah Microsoft Word 2000 untuk Windows.

- **Peralatan perisian untuk pembangunan**

Semasa proses pembangunan *Web Calculator*, beberapa perisian telah digunakan. Peralatan perisian yang digunakan di dalam fasa ini adalah seperti berikut:-

Jadual 6.2: Peralatan perisian untuk pembangunan

Perisian	Fasa/Proses
Microsoft Windows 98	Sistem pengendalian
Internet Information Server (IIS)	Hos pelayan web
JCreator LE	Mengkodkan applet.
J2SDK 1.4.1	Mencipta applet.
Macromedia Dreamweaver 4.0	Merekabentuk antaramuka
Adobe Photoshop 6.0	Merekabentuk <i>banner</i> .

6.4 Pengkodan

Pengkodan adalah proses menterjemah rekabentuk lengkap perwakilan perisian kepada bahasa pengaturcaraan yang sebenar.

6.4.1 Pendekatan Pengkodan

Kemahiran pengaturcaraan yang baik menghasilkan sistem yang boleh dipercayai dan mudah untuk diselenggarakan. Bentuk pengkodan yang baik memerlukan :-

- **Kebolehbacaan** – Kod dicipta dengan cara semudah yang mungkin di samping memastikan ia dapat mempersembahkan tugas yang diperlukan. Dengan cara ini ia lebih mudah difahami. Pengumpulan pengkodan membantu kepada kebolehbacaan. Kod sumber perlu boleh dibaca oleh pengaturcara yang lain dan juga bukan pengaturcara tanpa sebarang kesukaran. Ini memerlukan:-
 - pemilihan pencam (pembolehubah dan nama label)
 - kandungan komen
 - organisasi keseluruhan program
- **Teknik penamaan yang baik** – Ini bermakna nama yang diberi kepada pembolehubah, kawalan dan modul perlu mudah dikenalpasti oleh pengaturcara. Pengumpulan penamaan perlu dicipta dengan kekonsistenan dan piawaian pengkodan.

- **Dokumentasi dalaman** – Dokumentasi dalaman mengandungi maklumat yang ditujukan kepada sesiapa yang akan membaca kod sumber program. Oleh itu, maklumat rumusan disediakan untuk mengenalpasti program dan menggambarkan struktur data, algoritma dan aliran kawalannya. Kebiasaannya, maklumat ini diletakkan pada permulaan setiap komponen di dalam set komen yang dipanggil kepala blok komen (*header block comment*). Pengkodan di dalam *Web Calculator* menyediakan dokumentasi dalaman, maka pengaturcara lain mudah memahami kod tersebut.
- **Kemodularan** – Di dalam rekabentuk modular, komponen telah menakrifkan dengan jelas input dan output dan setiap komponen mempunyai tujuan yang jelas. Maka, ia mudah untuk menyemak setiap komponen secara berasingan dari yang lain untuk mengenalpasti sama ada komponen melaksanakan tugasnya. Tambahan pula, komponen modular diatur di dalam hirarki, sebagai hasil pengabstrakan, maka satu tahap sistem dapat disemak pada satu masa. Oleh sebab itu, perisian cuba direka menjadi modular.

Komponen perlu disusun di dalam tahap pengabstrakan yang berbeza. Tahap pengabstrakan membantu dalam memahami masalah sistem dan penyelesaian yang dicadangkan oleh rekabentuk. Dengan menyemak tahap-tahap dari atas dan kemudian ke bawah, masalah yang lebih abstrak akan dikawal

dahulu dan diikuti penyelesaiannya setelah gambaran lengkap telah dijanakan.

Kemodularan juga menyembunyikan perincian. Kelebihan penyembunyian maklumat ialah setiap komponen menyembunyikan keputusan rekabentuk dari yang lain. Maka, jika keputusan rekabentuk perlu diubah, keseluruhan rekabentuk akan kekal sementara hanya rekabentuk komponen sahaja yang berubah.

Pengabstrakan dan penyembunyian maklumat membenarkan semakan terhadap komponen yang mana berkait antara satu sama lain di dalam keseluruhan rekabentuk. Kebanyakan rekabentuk direka komponen yang tidak bersandar antara satu sama lain.

Untuk mengecam dan menyukat darjah ketidakbersandaran di dalam rekabentuk, dua konsep digunakan:-

1. **Coupling** – Dua komponen dikatakan *highly coupled* apabila terdapat kebersandaran yang kuat antaranya. Komponen *loosely coupled* mempunyai sedikit kebersandaran tetapi hubungan antara komponen adalah lemah. Komponen *uncoupled* tidak mempunyai hubungan langsung, ia sepenuhnya tidak bersandar.
2. **Cohesion** – Sebaliknya, untuk menyukat kebersandaran komponen, *cohesion* merujuk kepada “gam” dalaman yang

mana satu komponen dibina. Lebih *cohesion* sesuatu komponen itu, lebih berkait bahagian dalaman komponen antara satu sama lain dan kepada tujuan keseluruhannya. Di dalam kata lain, satu komponen adalah *cohesive* jika semua elemen komponen adalah menghala terus dan penting dalam melakukan tugas yang sama.

- **Pengumpulan pengkodan** – Pengumpulan pengkodan seperti melabelkan halaman, pengumpulan penamaan dan pemerenggan juga termasuk di dalam kategori ini.
- **Menyemak dan Gunasemula** – Fungsi dan prosedur yang direka lebih awal diguna semula di dalam modul berikutnya dengan menggunakan cara '*copy and paste*'.
- **Kebolehselenggaraan** – Kod perlu senang disemak dan dibetulkan. Untuk memudahkan penyelenggaraan, kod perlu mudah dibaca, modular dan seumum yang mungkin.

6.4.2 Gaya Pengkodan

Gaya pengkodan adalah atribut kod sumber yang penting dan ia menentukan kepintaran program. Kod sumber yang mudah dibaca menjadikan sistem lebih mudah untuk diselenggara dan ditambah. Elemen gaya termasuk dokumentasi dalaman (peringkat kod sumber), cara untuk meisytiharkan data dan pendekatan kepada pembinaan pernyataan.

Berikut adalah senarai beberapa gaya yang digunakan semasa pengkodan *Web Calculator*:-

1. Pemilihan nama pencam (pembolehubah dan label) yang bermakna.
2. Penerangan dan komen yang sesuai, ditulis di dalam kod sumber.
3. Pemerenggan kod meningkatkan kebolehbacaan kod sumber.

6.4.3 Pengkodan *Web Calculator*

Bahasa yang digunakan untuk membangunkan *Web Calculator* adalah Java.

Pengkodan sistem ini tidaklah begitu sukar kerana kalkulator dibina menggunakan java applet. Ini bermakna kod HTML dan kod java terasing atau tidak bercampur di dalam satu fail. Ini memudahkan aktiviti pengkodan dijalankan.

6.5 Metodologi yang Digunakan

Metodologi yang digunakan untuk pengkodan sistem ini adalah pendekatan atas bawah di mana modul peringkat tinggi dikodkan dahulu dan modul peringkat lebih rendah dibiarkan untuk dikodkan kemudian. Pendekatan ini digunakan untuk membenarkan pengujian dimulakan pada sesetengah modul yang telah siap sementara yang lain masih sedang dikodkan.

Pengujian sistem adalah serangkaian kegiatan dan prosedur pengujian yang dilakukan untuk memastikan bahwa sistem yang sedang dikembangkan memenuhi persyaratan yang telah ditetapkan dalam spesifikasi dan dokumen-dokumen lainnya yang dibuat. Selain itu, pengujian sistem juga bertujuan untuk memastikan bahwa sistem yang dikembangkan dapat beroperasi dengan baik dan sesuai dengan kebutuhan pengguna.

Pengujian sistem dilakukan pada berbagai tingkat, mulai dari pengujian unit, pengujian integrasi, pengujian sistem, hingga pengujian penerimaan. Setiap tingkat pengujian memiliki tujuan dan metode yang berbeda-beda. Pengujian sistem dilakukan untuk memastikan bahwa sistem yang dikembangkan dapat beroperasi dengan baik dan sesuai dengan kebutuhan pengguna.

PENGUJIAN SISTEM

Salah satu tujuan utama dari pengujian sistem adalah untuk memastikan bahwa sistem yang dikembangkan dapat beroperasi dengan baik dan sesuai dengan kebutuhan pengguna. Pengujian sistem dilakukan untuk memastikan bahwa sistem yang dikembangkan dapat beroperasi dengan baik dan sesuai dengan kebutuhan pengguna.

Pengujian sistem dilakukan pada berbagai tingkat, mulai dari pengujian unit, pengujian integrasi, pengujian sistem, hingga pengujian penerimaan. Setiap tingkat pengujian memiliki tujuan dan metode yang berbeda-beda. Pengujian sistem dilakukan untuk memastikan bahwa sistem yang dikembangkan dapat beroperasi dengan baik dan sesuai dengan kebutuhan pengguna.

7.1 Pengenalan

Pengujian sistem merujuk kepada penilaian dan pengesahan pengkodan program untuk menyelesaikan masalah. Penilaian melibatkan pemastian ciri-ciri rekabentuk yang baik disatukan ke dalam program dan sistem tersebut berfungsi seperti yang dijangka. Selain itu, pengesahan digunakan untuk menguji pelaksanaan program dan sistem memenuhi keperluan.

Pengujian sistem penting dalam proses pembangunan sistem. Ia adalah proses pelaksanaan program dengan mencari pepijat (*bugs*), ralat atau kecacatan yang wujud di dalam sistem. Pengujian sistem juga boleh ditakrifkan sebagai proses menganalisa item perisian untuk mengesan perbezaan antara keadaan yang wujud dan diperlukan dan untuk menilai ciri-ciri item perisian.

Selepas proses pengkodan atau pelaksanaan selesai, fasa pengujian akan dimulakan. Pada peringkat ini, pelbagai jenis pengujian dilakukan untuk memastikan produk akhir atau sistem akhir akan berfungsi seperti yang sepatutnya. Pelbagai jenis pengujian perlu dilakukan sebelum sistem dikeluarkan kepada pelanggan untuk memastikan sistem tersebut dibangunkan berdasarkan kepada spesifikasinya dan setiap fungsi yang dilaksanakan di dalam program berfungsi dengan betul.[4]

Pengujian bukanlah tempat pertama di mana "*faultfinding*" berlaku, tetapi pengujian memfokuskan kepada mencari kesalahan dan terdapat pelbagai cara untuk menjadikan pengujian lebih cekap dan efektif. Pengujian yang tidak mencukupi akan meninggalkan ralat, yang mana akan menyebabkan pertambahan kos, memberi tanggapan buruk kepada pelanggan dan pengguna dan pembetulan perlu dilakukan. Walau bagaimanapun, terlalu banyak pengujian

juga akan meningkatkan kos, namun ia masih tidak menjamin bebas dari ralat. Ia adalah kesedaran pengujian, bergabung dengan integrasi dan kesedaran dasar “pengaturcaraan berorientasikan-ujian” yang mana memberikan keyakinan di dalam kualiti produk semasa ia dihantar.

7.2 Objektif Pengujian

Tujuan pengujian adalah untuk mendedahkan kewujudan ralat di dalam perisian. Ia menunjukkan ujian yang baik mempunyai kemungkinan yang tinggi untuk menjumpai ralat. Pengujian yang berjaya akan menghasilkan perisian yang berkualiti dengan sedikit ralat dan berfungsi berdasarkan kepada spesifikasi dan persembahan keperluan. Pengujian boleh mendedahkan kelas-kelas ralat yang berbeza dengan jumlah masa dan usaha yang minimum.

Pengujian adalah penyukatan kawalan kualiti major yang bertindak untuk memastikan program dilaksanakan dengan betul dan sah mengikut keperluan yang ditetapkan. Ia membekalkan metod untuk pengesanan dan pembuangan ralat di samping untuk menguji kebolehpercayaan sistem.

7.3 Konsep Pengujian

Berdasarkan kepada kajian oleh Alka Jarvis, empat konsep asas berkait dengan pengujian sistem adalah:-

1. Pengesanan Ralat

- Melibatkan penyemakan dan *walkthroughs* untuk mengesan ralat.
- Pada peringkat unit, strategi pengujian berdasarkan kepada pengujian fungsian.
- Pada peringkat sistem, melibatkan pengujian fungsian atau pengujian kelakuan. (*behavioral*)

2. Pembuangan Ralat

- Melibatkan penyahpijat dan strategi untuk mengenalpasti di mana ralat berlaku di dalam kod dan membuangnya.
- Strategi seperti menjejak, memasang perangkap, mengenalpasti “keadaan ralat” dan mengenalpasti algoritma, fungsi atau modul yang mana ralat berlaku.

3. Menjejak Ralat

- Untuk mencari dan membetulkan penyebab ralat dan untuk membetulkan ralat itu sendiri.

4. Pengujian Regrasi

- Pengujian untuk melihat jika pembetulan atau *rework* kepada kod sebenarnya membetulkan ralat, betulkannya di dalam satu tempat dan pecahkannya ke lain, atau pecahkan kod di dalam tempat lain tanpa membetulkannya pada titik di dalam perisian di mana pembetulan cuba dilakukan.

7.4 Asas Pengujian

Pengujian adalah proses membuktikan kewujudan ralat. Perisian selalunya diuji mengikut peringkat. Pertama, setiap modul diuji secara berasingan dan ralat dibuang. Ini dinamakan pengujian modul atau unit. Modul yang diuji secara individu diintegrasikan dan diuji untuk memastikan antaramuka berfungsi dengan baik, dikenali sebagai pengujian integrasi. Pengujian fungsian adalah untuk menguji fungsian perisian supaya ia berfungsi seperti yang diperlukan. Kemudian, sistem diuji untuk pelaksanaan (pengujian pelaksanaan). Pelanggan akan melakukan ujian penerimaan sebelum mereka menerima sistem secara formal. Jika persekitaran pelanggan berbeza, ujian pemasangan perlu dilakukan. Rujuk rajah 7.1.

Secara umumnya, langkah-langkah di dalam pengujian adalah:-

1. Pengujian modular/unit
 - Pengujian unit untuk memastikan unit-unit dikodkan dengan betul dan ia menghasilkan fungsi yang betul

2. Pengujian integrasi

- Menguji satu kumpulan modul yang telah diintegrasikan (subsistem) atau keseluruhan sistem.
- Matlamatnya adalah untuk menentukan sistem atau subsistem mencapai keperluan sistem dan berfungsi dengan baik dan untuk menguji antaramuka antara modul.

3. Pengujian fungsian

- Ia diuji jika semua fungsian yang diperlukan oleh aplikasi berfungsi dengan baik.

4. Pengujian pelaksanaan

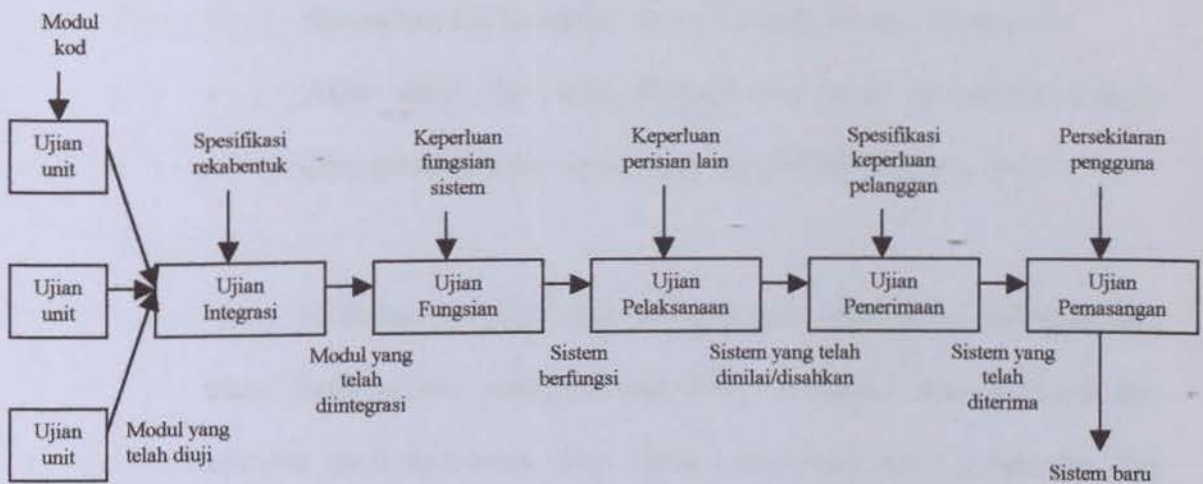
- Menguji pelaksanaan sistem mencapai spesifikasi yang diperlukan seperti keperluan bukan fungsian.

5. Pengujian penerimaan

- Diuji dengan data sebenar di dalam persekitaran yang sebenar. Ini bermakna sistem bersedia untuk digunakan.

6. Pengujian pemasangan

- Diuji jika sistem tersebut berfungsi dengan betul di dalam persekitaran yang sebenar.



Rajah 7.1: Peringkat-peringkat Pengujian

7.5 Strategi Pengujian

Strategi pengujian yang digunakan semasa pembangunan *Web Calculator* terdiri daripada pengujian unit, pengujian integrasi dan pengujian sistem.

7.5.1 Pengujian Unit

Di dalam peringkat pertama pengujian, setiap komponen program diuji bersendirian, diasingkan daripada komponen sistem yang lain. Pengujian unit mengesahkan komponen-komponen tersebut berfungsi dengan baik dengan jenis output dijangka daripada rekabentuk komponen. Pengujian unit dilakukan di dalam persekitaran yang terkawal. Langkah-langkah yang diambil di dalam pengujian unit:-

- Pertama, kod diperiksa dengan membacanya dan cuba untuk mencari algoritma, data dan kesalahan sintaks.

- Kemudian, kod dikompil dan kesalahan sintaks dihapuskan.
- Akhir sekali, kes ujian dibangunkan untuk menunjukkan input ditukarkan kepada output yang diinginkan dengan betul.

Di dalam pengujian unit, setiap langkah diperiksa sekali pada satu masa. Semasa fasa pengujian unit *Web Calculator*, semua fungsi dan subrutin kecil dan besar diuji untuk menyemak ralat pengkodan dan logikal. Tiga langkah asas yang dinyatakan di atas diikuti semasa pengujian unit.

Perkara berikut telah diuji semasa pengujian unit:-

- **Antaramuka** – uji antaramuka untuk memastikan maklumat mengalir ke dalam dan ke luar unit program dengan baik.
- **Analisis nilai sempadan** – memastikan modul beroperasi dengan baik pada sempadan.
- **Error handling path** – memastikan modul yang spesifik melaksanakan proses pemulihan apabila ralat berlaku.
- **Semua laluan program tidak bersandar telah dilaksanakan** – memastikan struktur kawalan dilaksanakan dengan betul.

7.6 Pengujian Integrasi

Pengujian integrasi adalah proses untuk mengesahkan komponen sistem berfungsi bersama seperti mana yang digambarkan di dalam spesifikasi “rekabentuk sistem dan program”. Ia memastikan antaramuka antara komponen-komponen di dalam sistem ditakrifkan dan dikawal dengan baik.

Di dalam *Web Calculator*, strategi ini melibatkan penggabungan modul satu demi satu, iaitu menggunakan pendekatan integrasi tokokan (*incremental integration*). Maka, sistem yang dibina diuji di dalam segmen yang kecil di mana ralat lebih cenderung untuk berlaku kerana ia lebih mudah diasingkan dan dibetulkan dan pendekatan pengujian yang sistematik mungkin akan digunakan. Tambahan pula, antaramuka juga perlu diuji dengan lengkap.

Pendekatan integrasi bawah-atas juga digunakan di dalam pengujian *Web Calculator*. Pada mulanya, setiap komponen pada peringkat paling bawah di dalam hirarki sistem diuji secara individu. Kemudian, komponen berikutnya yang akan diuji adalah komponen yang akan memanggil komponen yang telah diuji sebelumnya.

7.7 Pengujian Sistem

Pengujian sistem sebenarnya adalah satu aktiviti yang lengkap untuk mencari kelemahan dan mengukur keupayaan sesuatu penilaian. Pengujian sistem adalah untuk memastikan sistem adalah berfungsi dengan baik di bawah sistem yang besar. Pengujian sistem dilakukan terhadap *Web Calculator* adalah untuk memastikan semua elemen sistem telah diintegrasikan dan menjalankan fungsi seperti yang diperlukan. Langkah berikut dilakukan di dalam menguji sistem:-

1. Ujian Fungsian

Pengiraan di dalam kalkulator disemak supaya tiada ralat berlaku dan semua pautan (*link*) diuji supaya laman yang betul dipaparkan.

2. Ujian Kecekapan

Membandingkan komponen yang telah diintegrasikan dengan keperluan sistem bukan fungsian iaitu menguji kepantasan dan kebolehpercayaan kalkulator.

3. Ujian Penerimaan

Sistem diuji dengan meminta beberapa responden menggunakan sistem ini dan pendapat dan komen mereka tentang sistem ini diambil.

4. Ujian Pemasangan

Sistem dimasukkan ke dalam *server* dan juga dimuat turun (*upload*) ke dalam *web-hosting* di dalam internet. Pengguna menguji sistem tersebut di dalam kedua-dua persekitaran ini. Ini adalah untuk memastikan sistem berfungsi dengan baik di dalam persekitaran yang sebenar.

7.8 Teknik Pengujian

7.8.1 Pengujian Kotak Putih

Dengan menggunakan metod pengujian kotak putih, kes ujian berikut boleh diambil:-

- Menjamin semua laluan bebas di dalam satu modul telah diuji sekurang-kurangnya sekali.
- Menguji semua keputusan logikal pada sudut betul atau salah.
- Melaksanakan semua gelung pada sempadannya dan di dalam had operasi.
- Menguji struktur data dalaman untuk memastikan kesahihannya.

7.8.2 Pengujian Kotak Hitam

Pengujian kotak hitam adalah untuk mencari ralat di dalam kategori berikut:-

- Fungsi yang salah atau yang hilang.
- Ralat antaramuka
- Ralat di dalam struktur data.
- Ralat persembahan.
- Ralat permulaan dan pengakhiran.

PERBINCANGAN

8.1 Pendahuluan

Bab ini akan memfokuskan kepada beberapa masalah yang dihadapi semasa menjalankan projek dan juga penyelesaian yang telah diambil untuk menyelesaikannya. Di samping itu, bab ini juga akan memasukkan penilaian sistem untuk mengenalpasti kekuatan dan penghadannya.

8.2 Kekuatan Sistem

Semasa analisis dan pembangunan sistem ini, beberapa kekuatan telah dikenalpasti. Berikut adalah kekuatan *Web Calculator*:-

1. Sokongan Platform

Web Calculator dibangunkan di atas platform Windows yang mana merupakan sistem pengendalian yang paling umum dan biasa digunakan. Walau bagaimanapun, sistem ini boleh beroperasi pada platform utama yang lain seperti UNIX kerana Java tidak bergantung kepada platform.

2. Antaramuka Mesra Pengguna

Antaramuka *Web Calculator* adalah ringkas dan senang untuk digunakan. Rekabentuk laman web ini menggabungkan teknik yang paling biasa digunakan oleh pengguna Windows. Ia akan menjadi sistem yang mudah digunakan kerana Internet dan WWW sesuatu yang biasa bagi beribu pengguna di luar sana. Laman web diatur dengan baik untuk memastikan pengguna boleh navigasi dengan lancar melalui laman web dengan hanya cara 'tunjuk dan klik'.

3. Kekonsistenan

Menu sentiasa berada di kiri laman web, maka ia mewujudkan kekonsistenan. Di samping itu, *Web Calculator* juga menggunakan warna yang sama di dalam setiap laman. Dengan ini, pengguna tidak akan keliru semasa dalam proses navigasi. Ini juga akan membantu pengguna untuk belajar tentang sistem di dalam masa yang singkat.

4. Masa Tindakbalas yang Cepat

Oleh kerana laman web direka dengan ringkas dan ciri-ciri yang ditanam di dalam applet juga ringkas, paparan laman web adalah pantas (masa tindakbalas adalah bergantung kepada kadar paket yang dihantar dan diterima di dalam rangkaian data umum). Di samping itu, imej bergrafik yang besar dihindarkan. Proses pengiraan juga adalah cepat. Jawapan akan segera terpapar setelah pengguna menekan butang '=' atau selesai memasukkan kiraan.

5. Panduan Pengguna

Web Calculator menyediakan panduan kepada pengguna tentang butang-butang yang terdapat di dalam kalkulator serta fungsinya.

6. Fungsi Muat Turun

Web Calculator membenarkan pengguna memuat turun kalkulator tersebut ke dalam komputer mereka. Ini akan memudahkan pengguna

yang tidak mempunyai kalkulator di dalam sistem pengendaliannya dan inginkan kalkulator yang sentiasa sedia ada.

8.3 Kekangan Sistem

Walaupun usaha pelaksanaan yang baik telah digunakan untuk membangunkan sistem ini, namun masih terdapat beberapa kekangan. Kekangan-kekangan ini akan diatasi pada masa hadapan. Berikut adalah kekangan *Web Calculator*:-

1. Kalkulator Terlalu Ringkas

Pengiraan yang disediakan di dalam kalkulator ini adalah terlalu ringkas. Tidak banyak fungsi pengiraan yang boleh didapati contohnya pengiraan yang lebih kompleks.

2. Kekangan Pelayar

Walaupun *Web Calculator* adalah sistem yang tidak bergantung kepada platform, namun ia memerlukan pelayar yang dapat menyokong Java untuk memaparkan applet kalkulator tersebut.

3. Manual Pengguna

Manual pengguna tidak disediakan bersama fail yang dimuat turun. Ini mungkin akan menyukarkan pengguna untuk merujuk.

4. Butang

Terdapat banyak butang yang tidak mempunyai apa-apa fungsi diletakkan pada kalkulator. Ini menghasilkan antaramuka yang tidak berapa baik.

5. Tanda Operasi Bahagi

Tanda (/) digunakan sebagai tanda operasi bahagi. Mungkin akan mengelirukan pengguna yang tidak biasa dengan tanda ini.

8.4 Masalah dan Penyelesaian

1. Kekurangan Pengalaman dalam Pembangunan Web

Masalah:

Pembangun tidak mempunyai pengalaman atau pengetahuan di dalam aplikasi berasaskan web. Kekurangan pengalaman dan pengetahuan telah membuktikan terdapat halangan pada permulaan. Pembangun perlu berusaha memahami konsep pengaturcaraan dan aplikasi web dan perbezaannya.

Penyelesaian:

Masalah ini boleh diatasi dengan berusaha keras dan meminta nasihat daripada rakan sekuliah yang menggunakan peralatan yang sama. Di samping itu, terdapat banyak bahan rujukan di pasaran untuk pembangun. Selepas pengetahuan dan kemahiran telah dibiasakan, semua berjalan dengan lancar.

2. Tidak Berpengalaman di dalam Bahasa Pengaturcaraan

Masalah:

Disebabkan pengetahuan pengaturcaraan di dalam Java adalah cetek, pembangun sukar mengatur struktur dan kod semasa proses pengkodan dan larian. Pembangun juga menghadapi masalah dalam menggunakan Java dalam membangunkan applet kalkulator.

Penyelesaian:

Pembangun perlu memperuntukkan masa untuk belajar dan memahami bahasa dengan membaca dokumentasi JDK dan buku rujukan yang ada. Perbincangan dengan rakan sekuliah terutamanya dengan mereka yang menggunakan bahasa pengaturcaraan yang sama banyak membantu. Cara lebih cekap adalah melalui "*trial and error*" semasa fasa pengkodan. Tambahan lagi, melayari internet dan melibatkan diri dalam forum di dalam internet juga banyak membantu.

3. Applet Tidak Berjaya Dipaparkan

Masalah:

Disebabkan sistem ini dibangunkan di dalam komputer peribadi, maka pemindahan fail perlu dilakukan ke dalam *server* setelah siap dilakukan. Semasa fasa pengujian, applet kalkulator tidak dapat dipaparkan walaupun Plugin J2RE telah dimuat turun.

Penyelesaian:

Kod java terpaksa disalin semula ke dalam *server*, disimpan di dalam *folder* yang betul (*folder* yang terdapat fail-fail html yang lain) dan dikompil serta dilarikan semula. Akhirnya applet dapat dipaparkan.

8.5 Peningkatan Masa Hadapan

Sistem yang telah dibangunkan mempunyai beberapa kekurangan. Maka, beberapa penambahan perlu dibuat kepada sistem di masa hadapan untuk menjadikan ia lebih baik. Berikut adalah beberapa cadangan untuk penambahan:-

1. Lebih Banyak Fungsian

Pada masa hadapan, lebih banyak fungsian akan dimasukkan ke dalam kalkulator. Ia tidak hanya dapat melakukan operasi asas malah dapat menjalankan pengiraan yang lebih saintifik dan kompleks.

2. Manual Pengguna

Manual pengguna akan disediakan sekali dengan kalkulator yang dimuat turun. Ini untuk memudahkan pengguna untuk merujuk.

8.6 Pengetahuan yang Didapati

Melalui keseluruhan fasa pembangunan *Web Calculator*, tidak dinafikan banyak pengetahuan telah didapati. Antaranya ialah:-

- Mempelajari peralatan perisian tambahan – Bahasa seperti Java dan perisian seperti JCreator LE, Macromedia Dreamweaver 4.0 dan Adobe Photoshop 6.0 digunakan untuk membina aplikasi web.
- Mengetahui proses membangunkan perisian – Keseluruhan proses membangunkan perisian telah dilaksanakan, bermula dari keperluan pengguna kepada analisis, rekabentuk sistem, pengkodan dan akhir sekali pengujian perisian.
- Kemahiran di dalam menulis dokumen – Mengetahui cara dan format yang betul untuk menulis dokumen.
- Kemahiran di dalam pengurusan masa – Sistem mampu disiapkan tepat pada masanya.

8.7 Kesimpulan

Projek telah mencapai objektifnya, membangunkan *Web Calculator* iaitu kalkulator di atas talian (*online*). Semua keperluan dapat dicapai dan dapat berfungsi dengan baik. Namun terdapat beberapa kekurangan dan kekangan yang masih didapati selepas pembangunan sistem ini. Kekurangan dan kekangan ini perlu dikurangkan atau dihapuskan untuk menjadikan sistem ini berfungsi dengan lebih baik dan mesra pengguna. Peningkatan juga harus dilakukan supaya kalkulator ini tidak terhad kepada pengguna tertentu sahaja malah untuk kumpulan pengguna yang lain.

APENDIKS A: BORANG PENGUJIAN UNIT

Pengujian Unit

1. Laman web dipaparkan dengan cepat..... ☐
2. Semua komponen/imej dipaparkan..... ☐
3. Navigasi dapat berfungsi dengan betul..... ☐
4. Applet dapat dipaparkan..... ☐
5. Applet dipaparkan dengan cepat..... ☐
6. Semua butang nombor berfungsi dengan baik..... ☐
7. Semua butang operasi berfungsi dengan betul..... ☐
8. Kalkulator menjalankan pengiraan dengan betul dan tepat..... ☐
9. Ralat dipaparkan apabila kesalahan pengiraan berlaku..... ☐
10. Menu bantuan disediakan..... ☐
11. Fail dapat dimuat turun..... ☐

APENDIKS B:

BORANG

PENERIMAAN

PENGUNA

Borang Penerimaan Pengguna

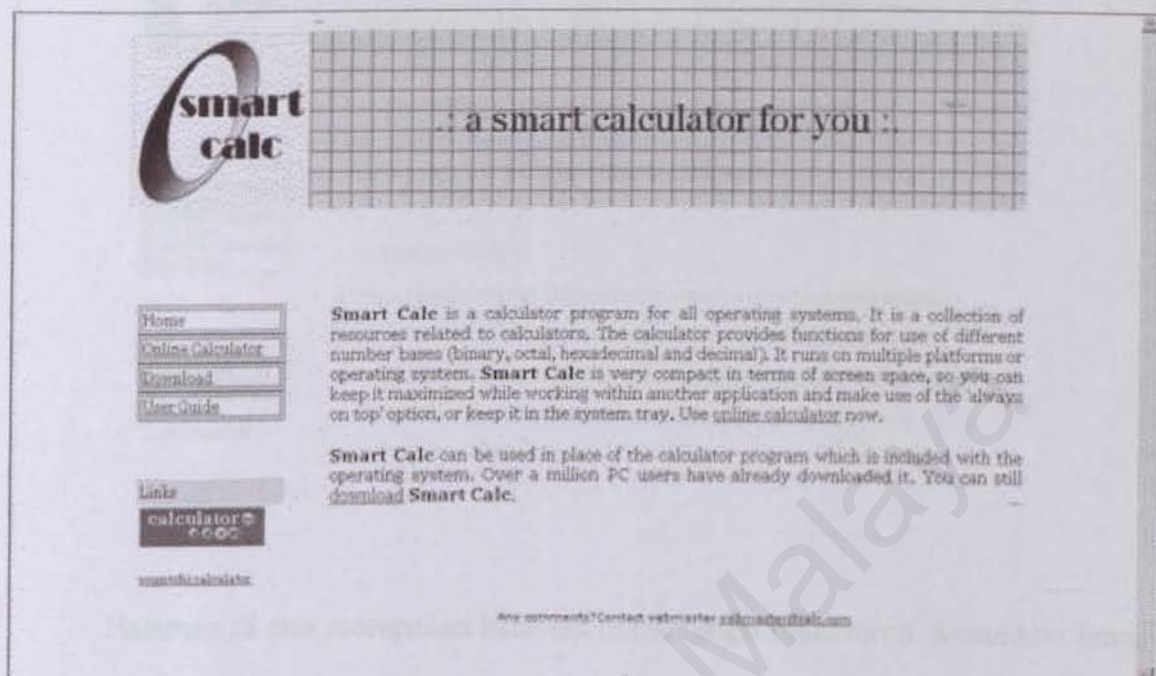
Sila tandakan (✓) di dalam kotak yang disediakan dan berikan komen/cadangan anda terhadap sistem ini.

	Ya	Tidak
1. Adakah antaramuka utama laman web ini menarik?	<input type="checkbox"/>	<input type="checkbox"/>
2. Adakah laman web ini mudah dilayari? (dari segi navigasi)	<input type="checkbox"/>	<input type="checkbox"/>
3. Adakah kalkulator dapat dipaparkan?	<input type="checkbox"/>	<input type="checkbox"/>
4. Adakah antaramuka kalkulator tersebut menarik?	<input type="checkbox"/>	<input type="checkbox"/>
5. Adakah kalkulator tersebut mengelirukan anda?	<input type="checkbox"/>	<input type="checkbox"/>
6. Adakah anda mudah mengendalikan kalkulator tersebut?	<input type="checkbox"/>	<input type="checkbox"/>
7. Adakah terdapat ralat di dalam pengiraan?	<input type="checkbox"/>	<input type="checkbox"/>
8. Adakah kalkulator menjalankan fungsi dengan baik?	<input type="checkbox"/>	<input type="checkbox"/>
9. Adakah anda berpuas hati dengan sistem ini?	<input type="checkbox"/>	<input type="checkbox"/>
10. Apakah komen/cadangan anda agar sistem ini dapat menjalankan fungsinya dengan lebih baik?		

Terima kasih atas kerjasama anda.

APENDIKS C:
MANUAL
PENGGUNA

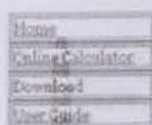
Manual Pengguna



Ini merupakan antaramuka utama. Terdapat butang pautan (*link*) di kiri tetingkap. Bahagian atas adalah pautan ke dalam laman web ini manakala bahagian bawah merupakan pautan ke laman web kalkulator yang lain.



... a smart calculator for you ...

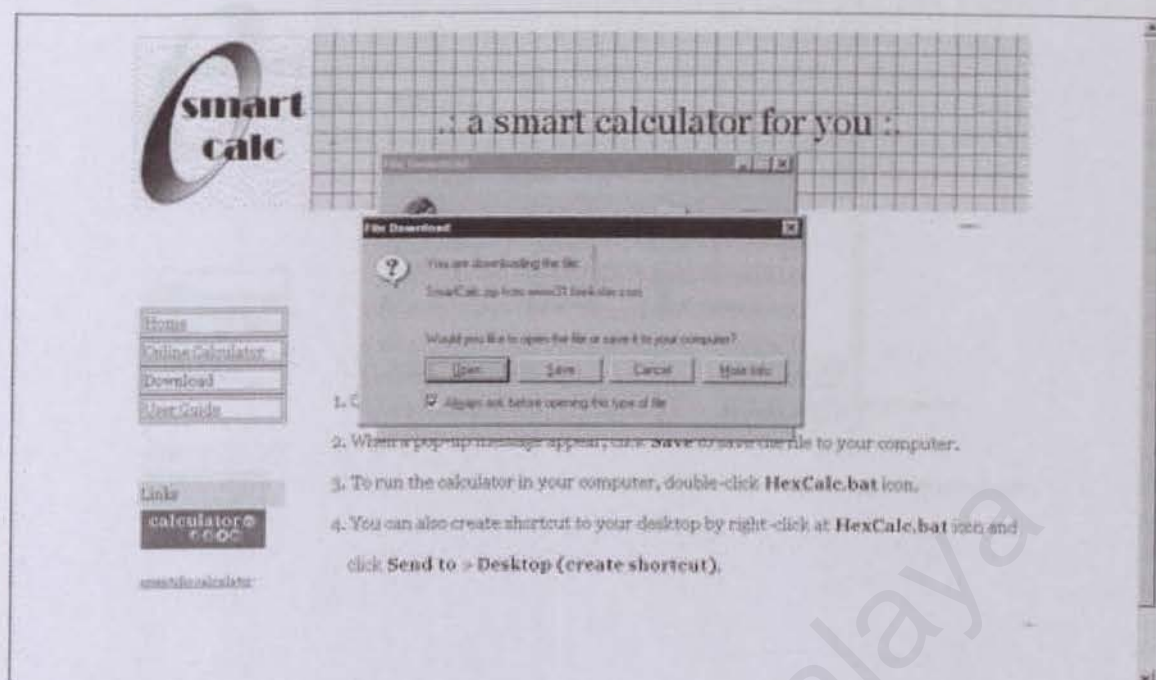


DOWNLOAD now!

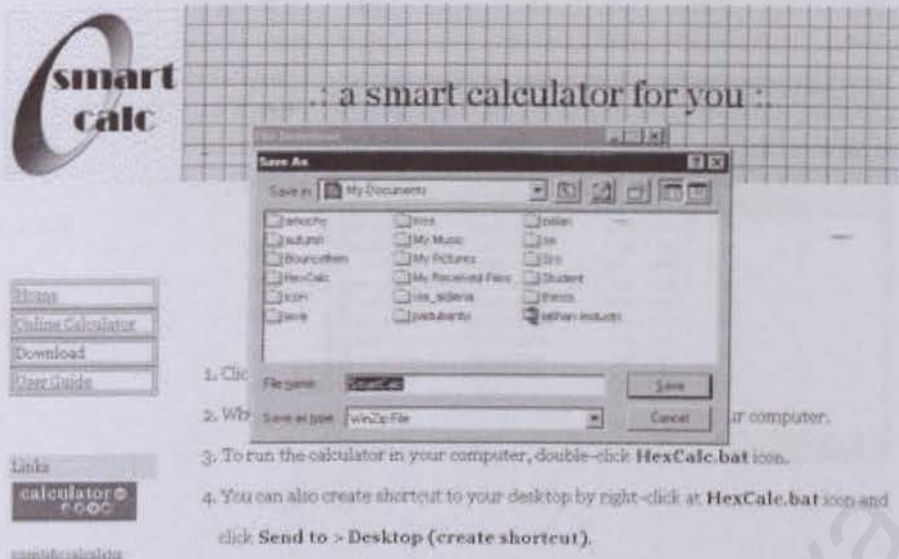


1. Click [here](#) to download.
2. When a pop-up message appear, click **Save** to save the file to your computer.
3. To run the calculator in your computer, double-click **HexCalc.bat** icon.
4. You can also create shortcut to your desktop by right-click at **HexCalc.bat** icon and click **Send to > Desktop (create shortcut)**.

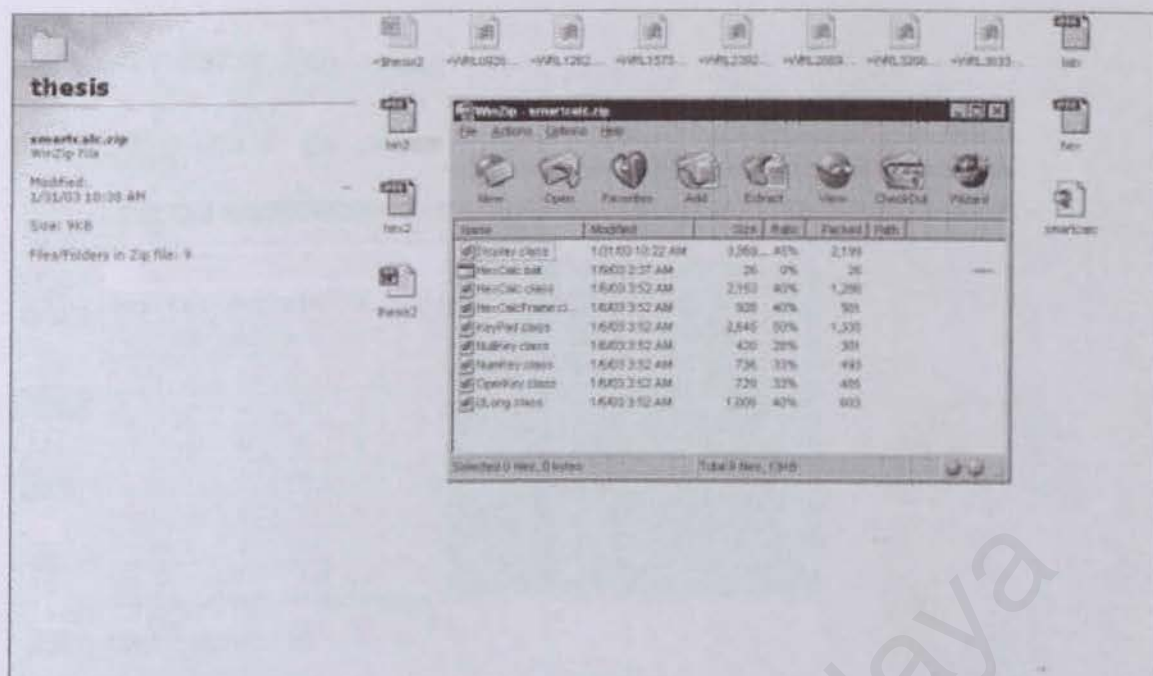
Halaman di atas merupakan halaman untuk fungsi muat turun. Susun atur laman adalah setara dalam semua halaman. Di bahagian tengah halaman memaparkan cara-cara untuk memuat turun kalkulator.



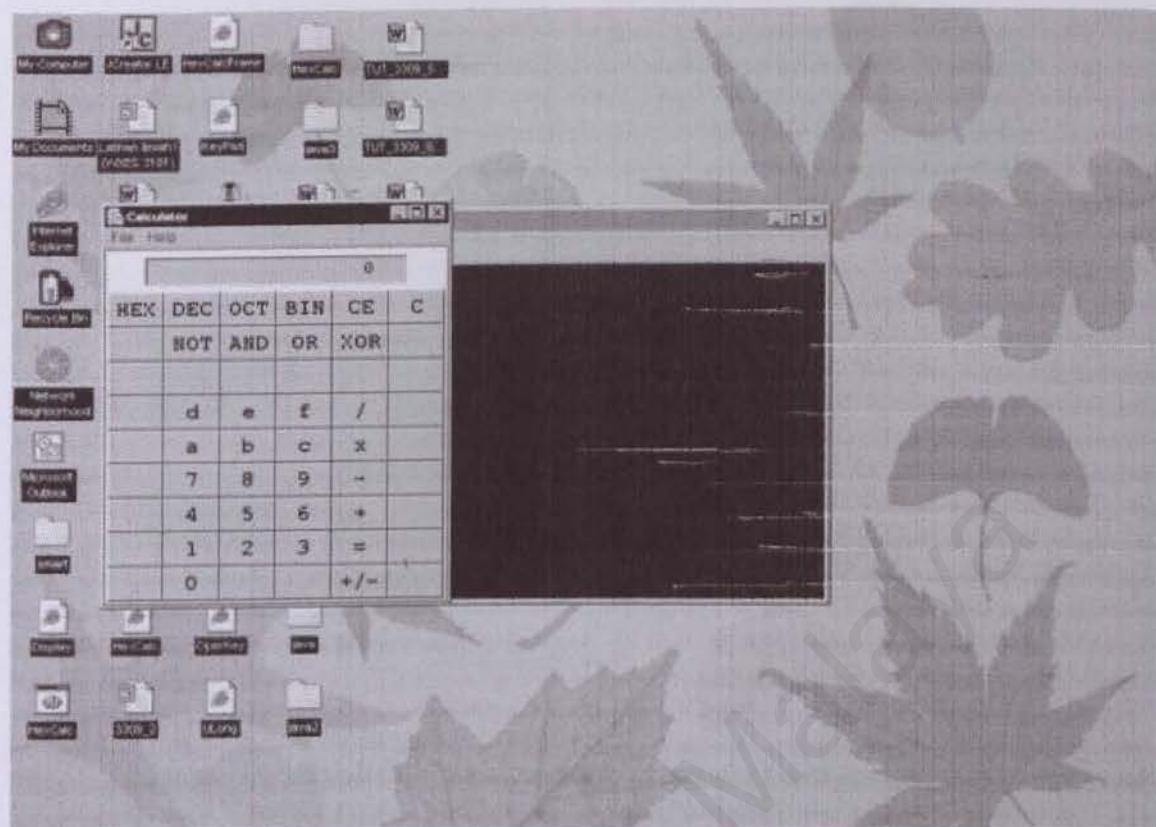
Apabila pengguna klik pada butang 'download', satu tettingkap seperti dalam rajah di atas akan dipaparkan. Pengguna perlu memilih butang 'Save'.



Apabila butang 'Save' telah diklik, satu paparan tetingkap yang lain akan terpapar. Tetingkap ini adalah untuk menentukan lokasi untuk menyimpan fail yang akan dimuat turun.



Fail yang dimuat turun adalah dalam bentuk *.zip. Maka fail tersebut perlu diekstrak.

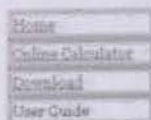


Setelah pengguna 'double-click' pada fail *.bat, paparan tetingkap akan dipaparkan seperti di atas.



... a smart calculator for you :

User Guide



Clear (C)

Clears the calculator, and resets any functions.

Clear Entry (CE)

Erases the last number or operation entered.

Addition (+)

This button performs addition.

Minus (-)

This button performs subtraction.

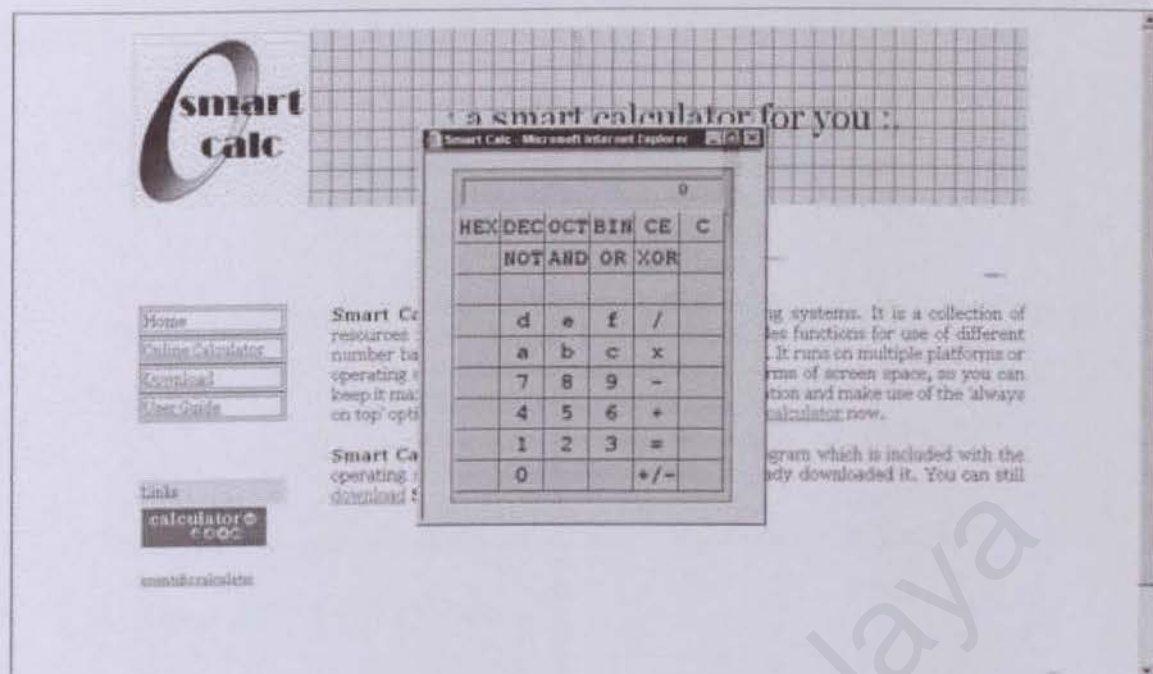
Divide (/)

This button performs division.

Multiply (x)

This button performs multiplication.

Halaman ini adalah halaman panduan pengguna (*user guide*). Susun atur tetap sama dan di bahagian tengah halaman memaparkan fungsi setiap butang di dalam kalkulator.



Apabila butang 'online calculator' diklik, satu applet akan terpapar. Saiz applet ini adalah kecil dan membolehkan pengguna melihat paparan di laman utama.

Rujukan

- [1] Dictionary of Computing, 4th Edition, Oxford University, 1995
- [2] Hayes, John R. (1978), *Computer for Architectural Organization*, 3rd Edition, McGraw-Hill.
- [3] Microsoft Windows, *Microsoft Windows 95*, Microsoft Corporation, 1995

RUJUKAN

- [1] Morgan, David (2001), *Microsoft Windows 95*, McGraw-Hill.
- [2] Morgan, David (2001), *Microsoft Windows 95*, McGraw-Hill.
- [3] Morgan, David (2001), *Microsoft Windows 95*, McGraw-Hill.
- [4] Morgan, David (2001), *Microsoft Windows 95*, McGraw-Hill.
- [5] Morgan, David (2001), *Microsoft Windows 95*, McGraw-Hill.

Rujukan

- [1] *Dictionary of Computing*, 4th Edition, Oxford University, 1996.
- [2] Hayes, John P. (1998), *Computer Architecture and Organization*, 3rd Edition, McGraw-Hill.
- [3] Mohamad Noorman Masrek et al, *Analisis Rekabentuk dan Sistem Maklumat*, McGraw-Hill (Malaysia), 2001.
- [4] Pfleeger, Shari Lawrence (2001), *Software Engineering Theory and Practice*, 2nd Edition, Prentice Hall.
- [5] Rodina Ahmad et al (2001), *Pengaturcaraan C++ Edisi Revisi*, McGraw-Hill (Malaysia).
- [6] Sellapan, P., *Programming in Java*, Sejana Publishing, 1st Edition, 1999.
- [7] Sufian Idris et al (2001), *OO.Java Pengaturcaraan Berorientasikan Objek Menggunakan Java*, McGraw-Hill (Malaysia).

BIBLIOGRAFI

Bibliografi

Kamus Perwira Bahasa Melayu, Bahasa Melayu – Bahasa Inggeris, Penerbitan Daya, Edisi Pertama, 1998.

Kamus Inggeris – Melayu Dewan, Dewan Bahasa dan Pustaka, Cetakan Pertama, 1991.

Deitel & Deitel, C++ How to Program, 3rd Edition, Prentice Hall.

Deitel & Deitel, Microsoft Visual J++, Prentice Hall.

Herbert Schildt, The Complete Reference Java 2, 4th Edition, McGraw Hill.

Liberty Jesse (1999), C++ from Scratch, Que Publishing.

Wai Chi, Liong (1998), A Networked Implementation of Othello (Nethello), Bachelor Thesis, University of Malaya

<http://www.java.sun.com>

<http://www.calculator.com/index.html>

<http://www.mrcalculator.com/hexdec.html>

<http://www.douze.net/hexa.php>

<http://www.ibilce.unesp.br/courseware/datas/numbers.htm>

<http://www.tpub.com/neets/book13/53g.htm>

<http://www.geocities.com/athens/parthenon/8960/osk6.html>

<http://www.cse.fau.edu/~maria/COURSES/CEN4010-SE/C13/black.html>

<http://www.csc.calpoly.edu/~sroliver/csc405/test/swtplan1.html>

University of Malaya